LUFP1
Telemecanique

User Manual

FIPIO / Modbus RTU
Gateway

Merlin Gerin
Modicon
Square D
Telemecanique

Schneider Electric

ENGLISH

NOTE

In spite of all the care taken over the writing of this document, Schneider Electric SA does not give any guarantees in relation to the information contained in it, and may not be held liable for any errors, nor for any damage which might result from its use or its application.

The characteristics and operation of the products and additives presented in this document may change at any time. The description is in no way contractually binding.

Contents

ENGLISH

1. Introduction

1.1. Introduction to the User Manual

The LUFP1 Gateway User Manual is organised into a series of chapters, which can in turn be grouped together into three parts as follows:

ENGLISH

- **Part I:** Covers all the elements required in order to set up the gateway and operate it on a FIPIO network.

  Chapter 1: Introduction (page 6) describes the gateway, the user manual that comes with it and the terms used in it.

  Chapter 2: Hardware Implementation of the LUFP1 Gateway (page 13) gives an introduction to the gateway and describes all the items used when setting it up, both inside the gateway (thumb wheels) and outside (cables and connectors).

  Chapter 3: Signalling (page 23) describes the six LEDs on the front of the gateway.

  Chapter 4: Software Implementation of the Gateway (page 25) describes the successive steps for setting the gateway up with a PLC using FIPIO. You need to use PL7 PRO to configure the gateway and thus interface a FIPIO master with Modbus slaves. The example used in this implementation enables the FIPIO master to dialog with 8 TeSys U motor starters.

  Chapter 5: FIPIO Objects Available for Programming (page 38) describes all the FIPIO objects associated with the LUFP1 gateway and which you can use with PL7 PRO.

1. Introduction

- **Part II:** The second part is dedicated to a third-party application, “ABC-LUFP Configurator”, which enables the gateway to be configured in greater detail than as set out in Part I. This part therefore supersedes Chapter 4: Software Implementation of the Gateway.

  Chapter 6: Advanced Implementation of the Gateway (page 56) describes an alternative method for configuring the LUFP1 gateway. Rather than using PL7 PRO, the gateway is configured with a third-party software known as “ABC-LUFP Configurator”, allowing for more detailed configuration of the LUFP1 gateway. The example used in this implementation is the same as in Chapter 4: Software Implementation of the Gateway (see above).

  Chapter 7: Using ABC-LUFP Configurator (page 61) describes how to use ABC-LUFP Configurator to create or modify a configuration for the gateway, but only in the context of an Advanced Implementation of the Gateway (Chapter 6). It also presents the various software functions (add or delete a Modbus slave, add or modify a Modbus command, etc.) and the changes to be made regarding advanced gateway implementation operations with PL7 PRO (Chapter 6).

- **Part III:** The third and final part contains the full set of Appendices which supplement the first two parts.

  Appendix A: Technical Characteristics (**Chapter 8**, page 96) describes the technical aspects of both the gateway and the networks it is interfaced with, namely the FIPIO and Modbus RTU networks.

  Appendix B: LUFP1 Gateway Settings (**Chapter 9**, page 100) describes the configuration and adjustment parameters for the LUFP1 gateway.

  Appendix C: Standard Configuration (**Chapter 10**, page 111) describes the main features of the configuration used in the Advanced Implementation of the Gateway (Chapter 6). However, it does not go into ABC-LUFP Configurator in detail.

  Appendix D: Sample Use under PL7 PRO (**Chapter 11**, page 114) provides an example of how the LUFP1 gateway is used, based on the configuration described in the Software Implementation of the Gateway (Chapter 4). This example employs the command and control registers for 8 TeSys U motor starters and uses the gateway’s indexed periodic variables (PKW) service and active slave list (LAS) service.

  Appendix E: Modbus Commands (**Chapter 12**, page 120) describes the contents of the Modbus command frames supported by the LUFP1 gateway.

1. Introduction

1.2. Introduction to the LUFP1 Gateway

The LUFP1 gateway allows a master located on a FIPIO network to enter into a dialogue with the slaves on a Modbus RTU network. This is a generic protocol converter operating in a way which is transparent to the user.

This gateway allows you to interface many products marketed by *Schneider Electric* with a FIPIO network. These include TeSys U motor starters, Altivar drivers and Altistart soft start / soft stop units.

1.3. Terminology

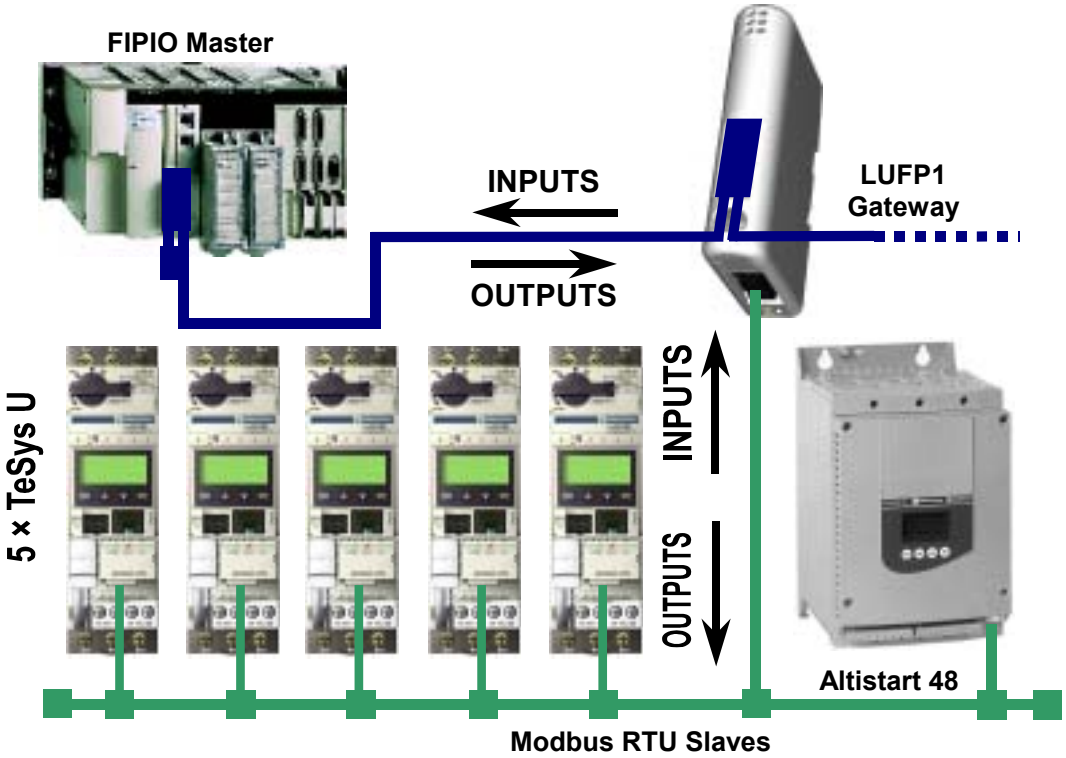
Throughout this document, the term “user” refers to any person or persons who may need to handle or use the gateway.

The term “RTU”, which refers to the Modbus RTU communication protocol, will be omitted most of the time. As a result, the simple term “Modbus” will be used to refer to the Modbus RTU communication protocol.

As is still the case with all communication systems, the terms “input” and “output” are somewhat ambiguous. To avoid any confusion, we use a single convention throughout this document. So the notions of “input” and “output” are always as seen from the PLC, or the FIPIO master.

Hence, an “output” is a command signal sent to a Modbus slave, whereas an “input” is a control signal generated by this same Modbus slave.

The diagram below shows the flows of “inputs” and “outputs” exchanged between a FIPIO master and Modbus RTU slaves via the LUFP1 gateway:



ENGLISH

1. Introduction

1.4. Notational Conventions

16#•••• ..... Value expressed in hexadecimal, which is equivalent to the H••••, ••••h and 0x•••• notations, sometimes used in other documents. **N.B.** The AbcConf software uses the 0x•••• notation: e.g. 16#0100 = 0x0100 = 256.

02#•••• •••• ..... Value expressed in binary. The number of '•' digits depends on the size of the item of data represented. Each nibble (group of 4 bits) is separated from the other nibbles by a space.

Examples: byte 2#0010 0111 = 39, word 2#0110 1001 1101 0001 = 16#69D1 = 27089.

AbcConf ..... Abbreviation that refers to the tool used to configure and implement the LUFP1 gateway: "ABC-LUFP Configurator".

ASIC ..... Integrated circuits specific to a given user and application, covering two major families: pre-characterised processes and pre-distributed networks.

ATS ..... Abbreviation of "Altistart" (soft start / soft stop unit).

ATV ..... Abbreviation of "Altivar" (drive).

CRC ..... Cyclical Redundancy Check.

LED ..... Light-Emitting Diode.

DWF ..... Device WorldFIP. Solution WorldFIP commonly known as FIPIO.

FED C32 ..... Term for one of the standard FIPIO station profiles. FED = extended profile; C = compact equipment; 32 = 32 periodic I / O variables (16-bit format). This is the profile that must be used for configuring the LUFP1 gateway in PL7 PRO, but *only* in the context of Advanced Implementation of the Gateway (Chapter 6, page 56).

FED C32 P ..... See FED C32 above. P = presence of parameters (30 configuration words and 30 adjustment words in this instance). *This is the profile that must be used for configuring the LUFP1 gateway in PL7 PRO in the context of the Software Implementation of the Gateway (Chapter 4, page 25).*

Fieldbus ..... A term referring to the upstream FIPIO under AbcConf.

FIP ..... Factory Instrumentation Protocol. An old term for the WorldFIP field bus.

FIPIO ..... WorldFIP Solution using the second WorldFIP bus communication profile. *Profile 1* is dedicated to sensors / actuators and other equipment that handles relatively little information and requires little parameterisation. *Profile 2* (FIPIO) is weakly configurable and parameterisable; the information is exchanged periodically, with the possibility of exchanges in aperiodic mode. As the LUFP1 gateway is a FIPIO slave, it conforms to WorldFIP bus *profile 2*.

FIPWAY ..... A "FieldBus WorldFIP" solution comprising the two latest WorldFIP bus communication profiles. *Profile 3* is configurable and parameterisable; the information is exchanged periodically and event-driven exchanges are authorised. *Profile 4* is reserved for complex equipment; it is configurable and parameterisable, and supports downloading.

LAS ..... List of Active Slaves. This service notifies the FIPIO master of the presence or absence of each one of the Modbus slaves in the LUFP1 gateway.

LRC ..... Longitudinal Redundancy Check.

Node ..... A term referring to the connection point of a Modbus slave under AbcConf.

LSB: ..... Least significant byte in a 16-bit word.

MSB: ..... Most significant byte in a 16-bit word.

PKW ..... Parameter Kennung Wert. German abbreviation used to designate the service that handles the indexed periodic variables of configuration and adjustment. This service offers read-write access to any parameter on any of the Modbus slaves via the periodic input and output of the LUFP1 gateways.

Sub-Network ..... A term referring to the downstream Modbus network under AbcConf.

WorldFIP ..... Name of the WorldFIP field bus, and of the association that brings together its suppliers and users.

XML ..... EXtensive Markup Language. The language used by AbcConf to import / export the configuration of Modbus slaves.

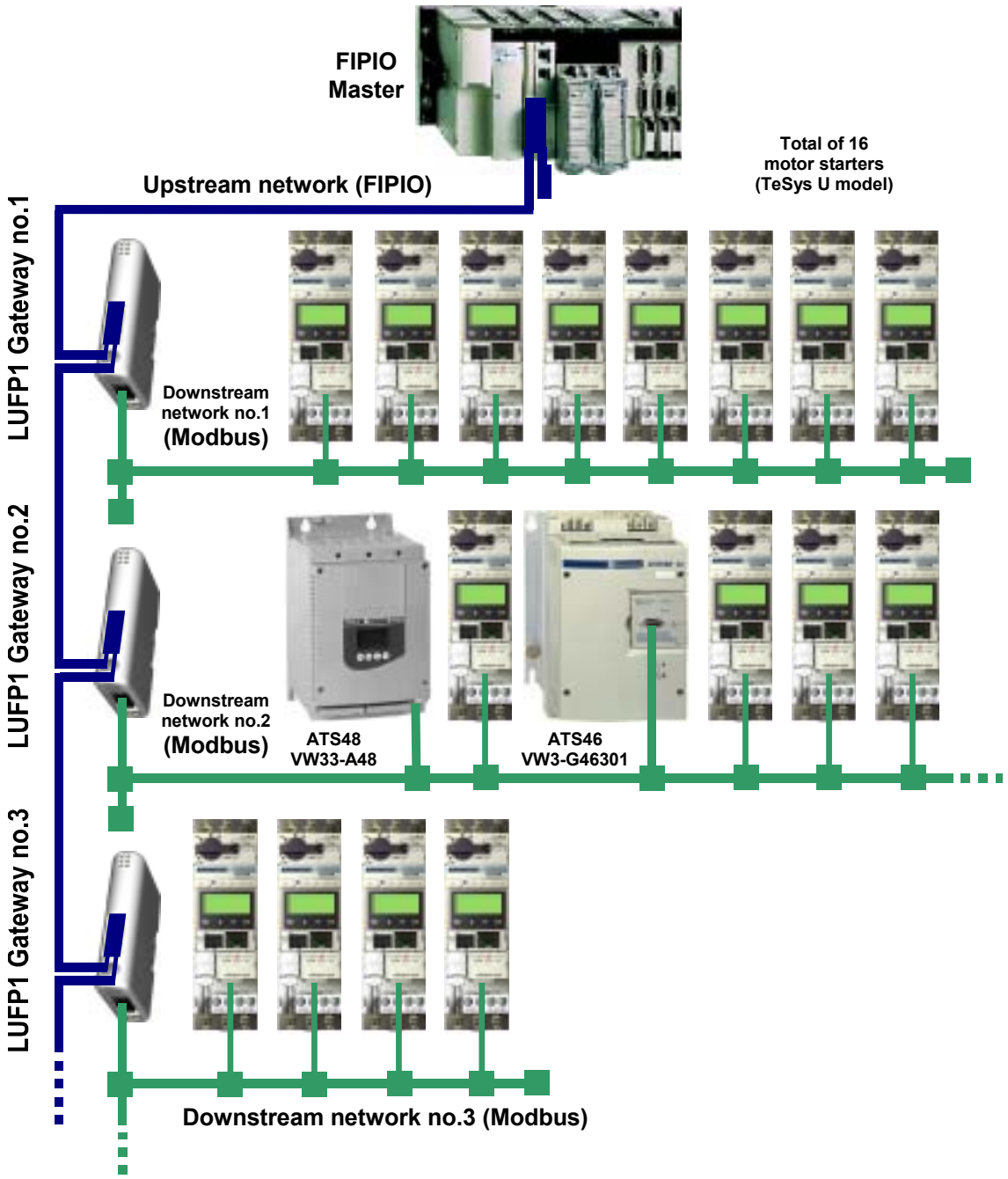
ENGLISH

1. Introduction

1.5. Additional Documentation

In the case of Modbus slaves, the features, services and adjustment of the Modbus communications are not dealt with in this document.

1.6. Introduction to the Communication "System" Architecture



ENGLISH

1. Introduction

Each LUFP1 FIPIO / Modbus RTU gateway allows one PLC on the FIPIO network to command, control and configure up to 8 Modbus slaves. If there are more than 8 Modbus slaves, you will need to use an appropriate number of LUFP1 gateways. Likewise, if the gateway needs to exchange more than 26 words with the Modbus slaves (sum of the read and write operations), the Modbus slaves will have to be shared over more than one gateway.

The LUFP1 gateway behaves both as a FIPIO slave on the upstream network and as a Modbus RTU master on the downstream network.

See Section 8.2 Communication Characteristics, page 96, if you would like to read about the technical communication characteristics of the LUFP1 gateway.

In the context of the Software Implementation of the Gateway, the data exchanges (input and output words) between the gateway and the Modbus slaves are all periodic. Collectively, these Modbus exchanges form the gateway's "Modbus scanner"; they are configured using PL7 PRO (by configuring and adjusting the standard profile FED C32 P). Every item of data exchanged in this way is made available to the FIPIO master, which can access it periodically and implicitly. The only aperiodic exchanges that can be performed with the LUFP1 gateway on the FIPIO network are explicit exchanges relating to the adjustment parameters and gateway status parameters.

The gateway has a configuration and adjustment service called PKW, which offers read-write access to any Modbus slave parameter via the periodic input and output of the LUFP1 gateways. This service is aperiodic on the Modbus network and periodic on the FIPIO network. It can used for the following tasks:

- Retrieving or updating the parameters on some or all of the Modbus slaves;
- Reading the value of one or more data for which a low refresh rate is sufficient;
- Reading the value of a data item when an event is signalled by a periodic variable.

One of the gateway input words informs the FIPIO master of the presence or absence of each Modbus slave. The input word in question is the LAS service.

The diagram on the left page illustrates the distribution of several slaves throughout three Modbus RTU downstream networks, each one being interfaced with the FIPIO master PLC using a LUFP1 gateway.

1.7. Principles of LUFP1 Gateway Configuration and Operation

The gateway is part of a family of products (referred to as LUFP●) designed to meet generic needs for connection between two networks using different communication protocols.

The software elements common to all these gateways (a configuration tool known as "ABC-LUFP Configurator" and the embedded Modbus software) cohabit with the specific features of the network upstream of each of them (FIPIO in the case of the LUFP1 gateway) generically. This is one of the reasons why the interfacing between the upstream network and the Modbus network is carried out entirely via the gateway's physical memory. In the case of the LUFP1 gateway, however, using a standard FIPIO profile (FED C32 P) in PL7 PRO makes it easier to configure the gateway, dispensing with the need to use ABC-LUFP Configurator.

⇨ Exchanges between the gateway (acting as the Modbus master) and the Modbus slaves are configured by the gateway, using the configuration and adjustment parameters of profile FED C32 P entered by the user in PL7 PRO. Thanks to this configuration, the gateway creates links between part of the contents of the corresponding Modbus frames and the contents of its physical memory (input memory for the contents of the Modbus responses and output memory for the contents of the queries).

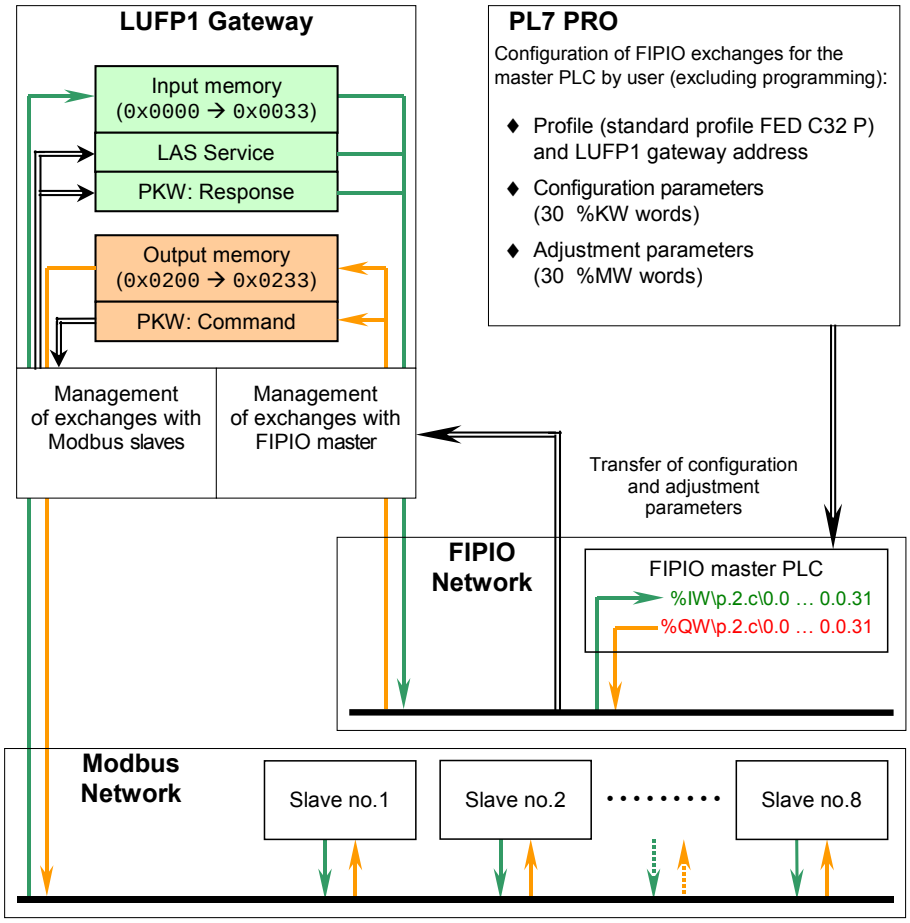
1. Introduction

- ⇨ The periodic exchanges between the FIPIO master PLC and the LUFP1 gateway are limited to transferring the contents of the gateway's input memory to the %IW inputs on the PLC and transferring the PLC's %QW outputs to the gateway's output memory.
- ⇨ Each LUFP1 gateway is delivered so that it can be configured using PL7 PRO instead of ABC-LUFP Configurator. This method makes the gateway easier to configure. However, the gateway is limited to a blank pre-configuration, as it will be configured by the master PLC itself when the FIPIO network is initialised.

The FIPIO network is totally separate from the Modbus network. The frames on a network are not directly "translated" by the gateway to generate frames on the other network. Instead, the exchanges between the contents of the gateway's memory and the Modbus slaves make up a system independent of the one entrusted with managing the exchanges between the gateway memory and the FIPIO master.

The following synopsis illustrates the independent management of each of the two networks:

— Management of FIPIO master ⟷ gateway ⟷ Modbus slave exchanges —



Syntax: p = processor position (0 or 1);
c = connection point number (gateway address).

2. Hardware Implementation of the LUFP1 Gateway

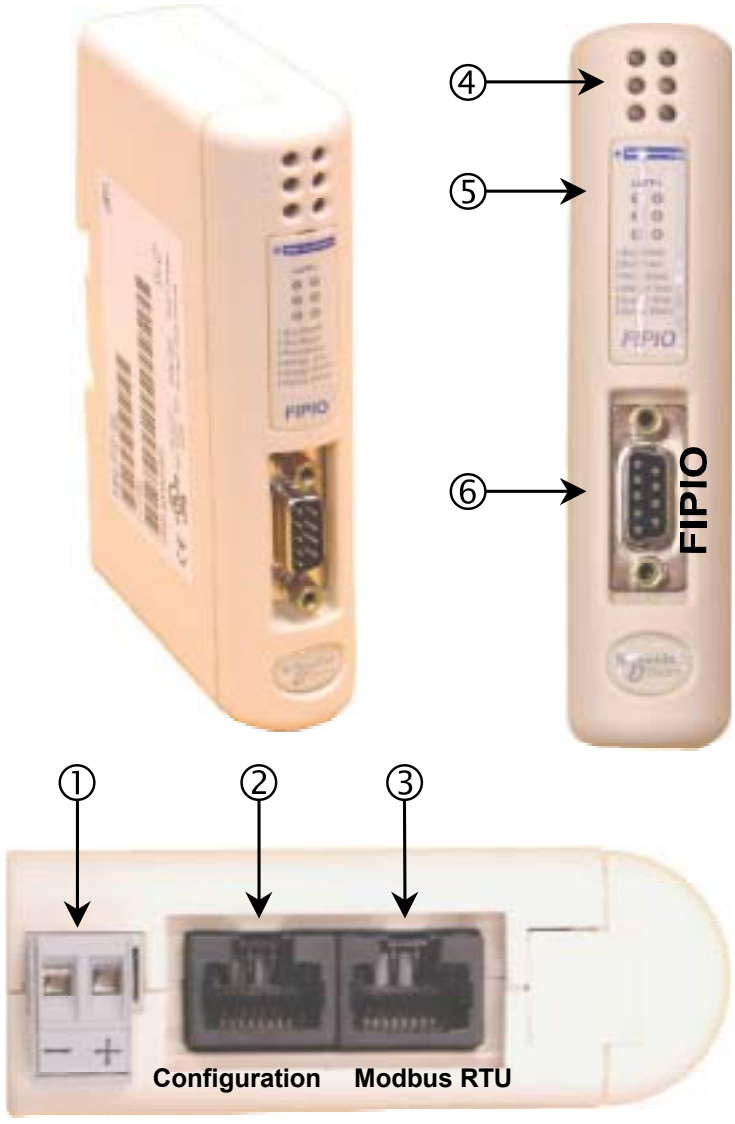
2.1. On Receipt

After opening the packaging, check that the following element is present:

- One LUFP1 FIPIO / Modbus RTU gateway.

2.2. Overview of the LUFP1 Gateway

The cables and other accessories for connecting to FIPIO and Modbus networks need to be ordered separately.



Legend:

① Detachable power connector for the gateway (⎓ 24V ±10%).

② Female RJ45 connector to a PC running AbcConf configuration software.

③ Female RJ45 connector for the downstream Modbus RTU network.

④ Six diagnostic LEDs.

⑤ Removable cover for the coding wheels used to configure the gateway, shown and described in Section 2.7 Configuring the FIPIO Communication Features, page 22. The label describing the LEDs is stuck onto this cover.

⑥ Male FIPIO connector.

ENGLISH

2. Hardware Implementation of the LUFP1 Gateway

2.3. Mounting the Gateway on a DIN Rail

Mounting the gateway

Start by fitting the rear base of the gateway to the upper part of the rail, pushing downwards (1) to compress the gateway's spring. Then push the gateway against the DIN rail (2) until the base of the gateway box fits onto the rail.

Removing the gateway

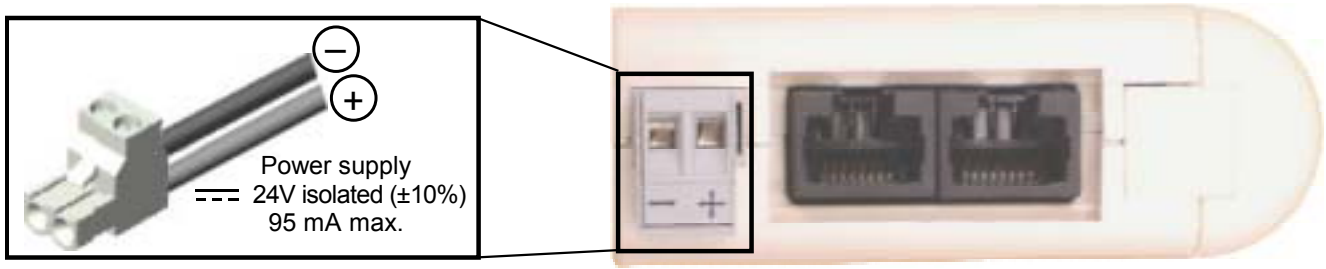
Start by pushing the gateway downwards (1) to compress the gateway's spring. Then pull the bottom of the gateway box forwards (2) until the box comes away from the rail.

N.B. The spring is also used to earth the gateway (Protective Earth).

ENGLISH

2.4. Powering the Gateway

FIPIO / Modbus RTU gateway – View from underneath



N.B. The negative 24V power supply terminal ⎓ should be connected to the installation's earth.

2. Hardware Implementation of the LUFP1 Gateway

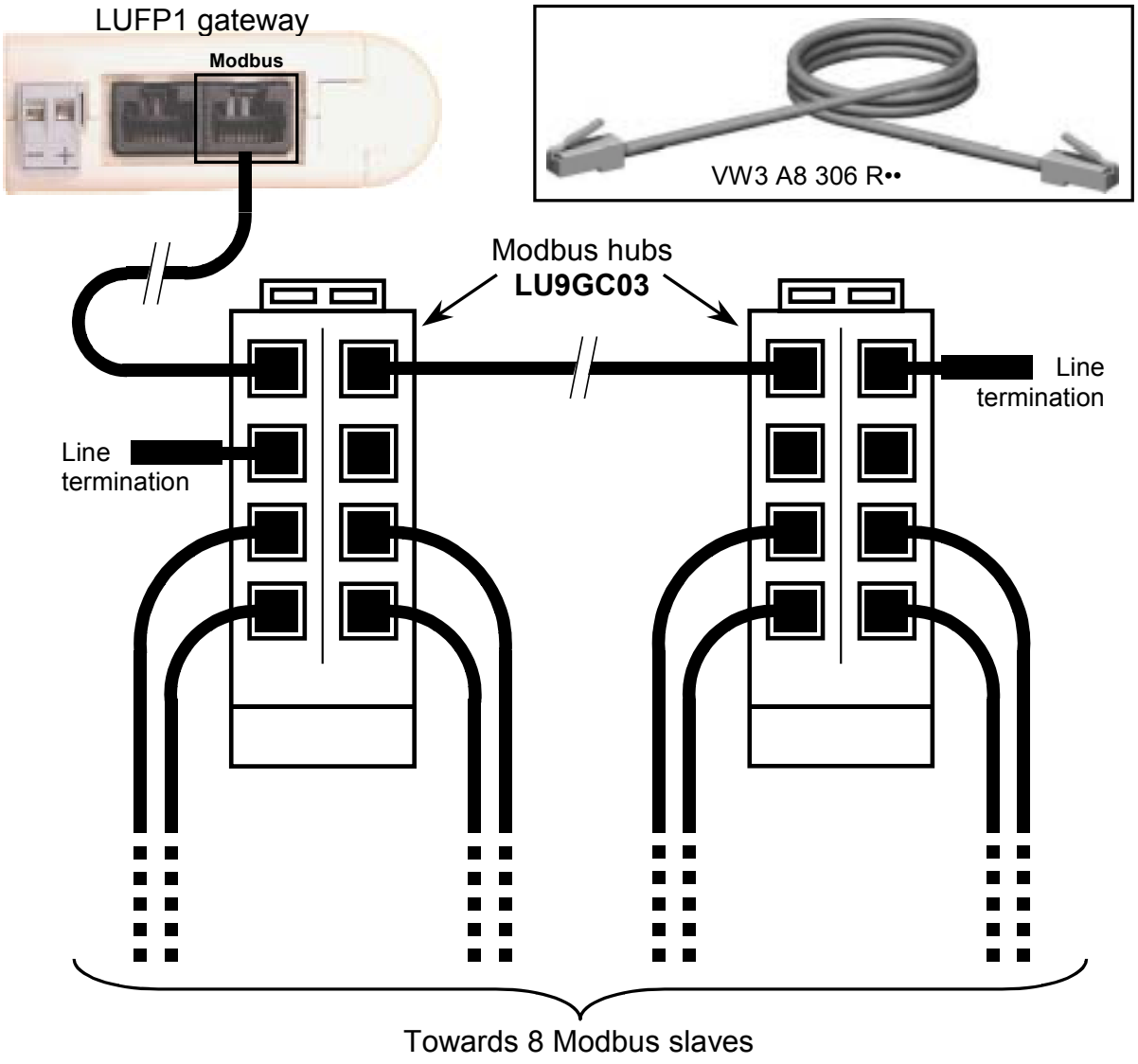
2.5. Connecting the Gateway to the Modbus Network

Three typical examples of Modbus connection for the gateway and its slaves are shown below. There are many other possible Modbus connections, but they are not covered in this document.

2.5.1. Examples of Modbus Connection Topologies

- **"Star" topology:** This topology uses LU9GC03 Modbus hubs, which have 8 female RJ45 connectors. These hubs should be placed close to the Modbus slaves to which they are connected using VW3 A8 306 R•• cables. On the other hand, the nature of the cable connecting the LUFP1 gateway to one of these hubs will depend on the network architecture, so long as there is a male RJ45 connector at each end. If necessary, one or two line terminations may be directly connected to the hubs.

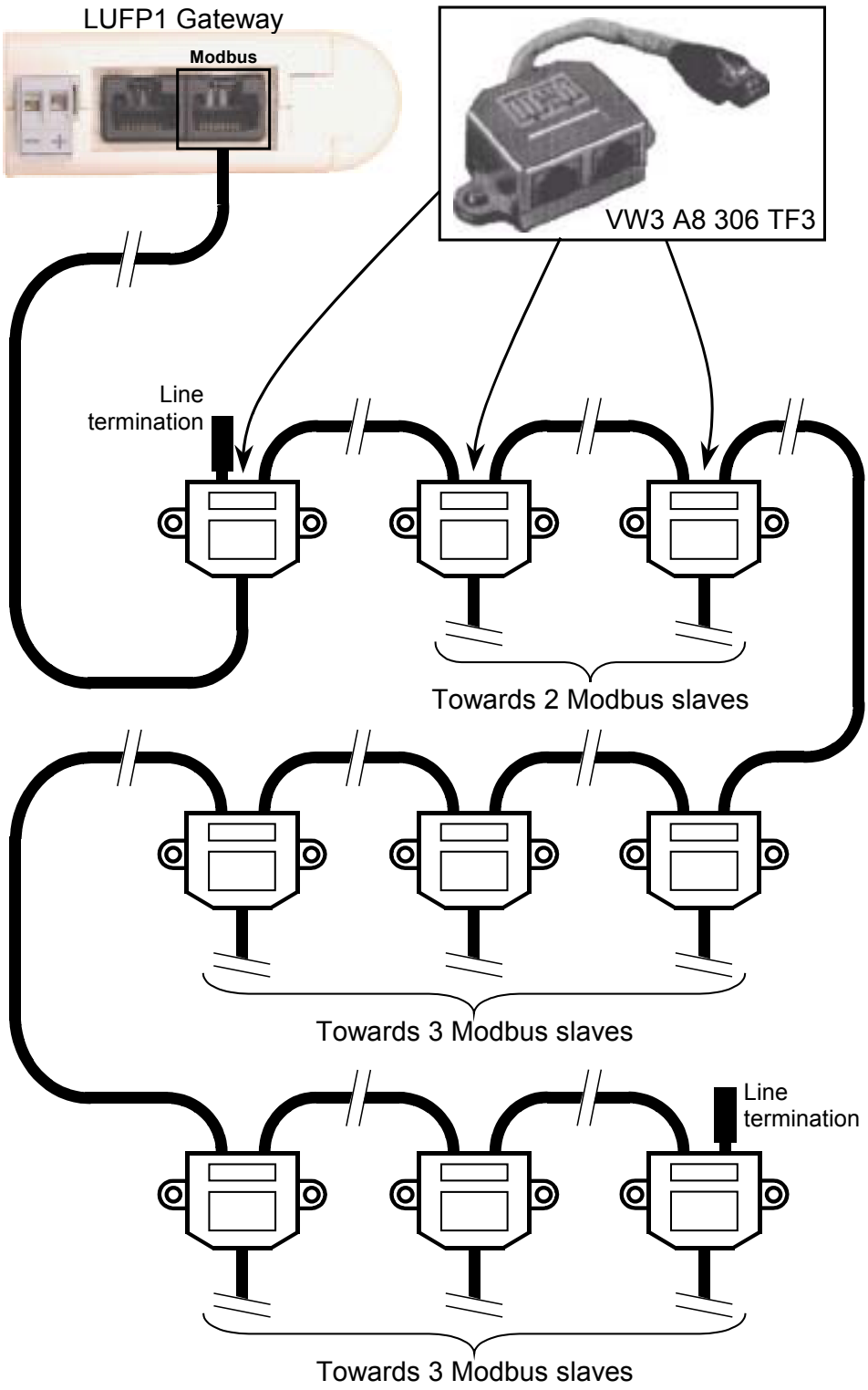
  The connections are shown below:



ENGLISH

2. Hardware Implementation of the LUFP1 Gateway

- **"Bus" topology with VW3 A8 306 TF3 drop boxes:** This topology uses VW3 A8 306 TF3 drop boxes to connect each of the Modbus slaves to the main section of the Modbus network. Each box should be placed in the immediate vicinity of the Modbus slave it is associated with. The cable for the main section of the Modbus network must have male RJ45 connectors (like the VW3 A8 306 R•• cable used for the "star" topology). The lead between the drop box and the slave or the Modbus gateway is an integral part of this box. The connections are shown below:

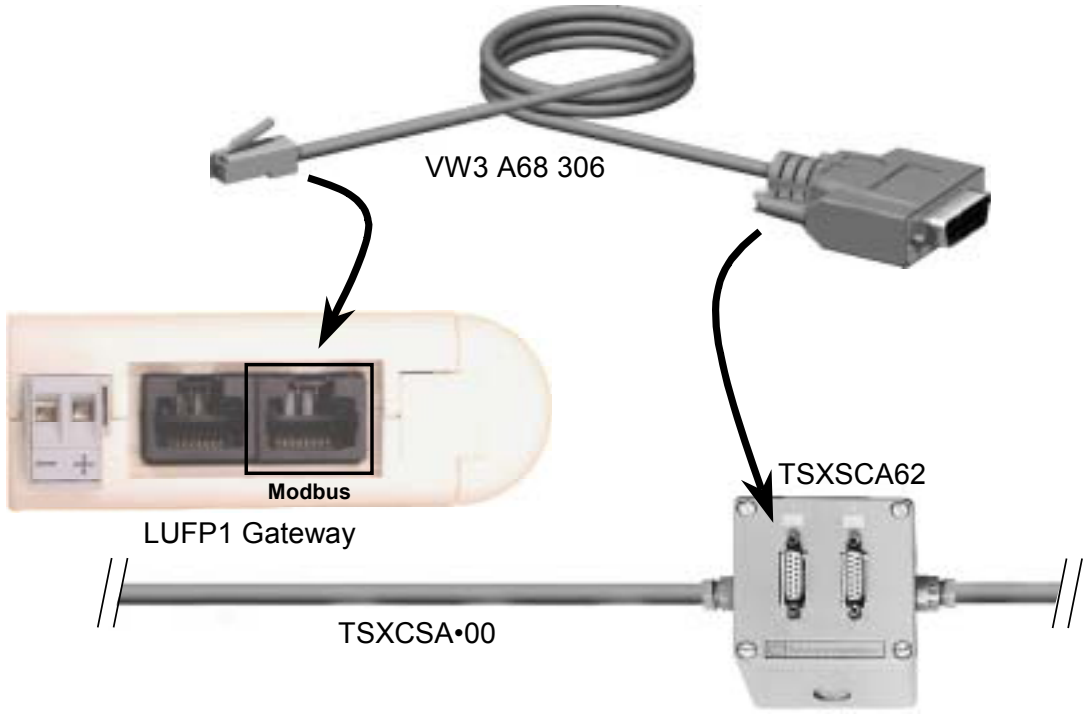


ENGLISH

2. Hardware Implementation of the LUFP1 Gateway

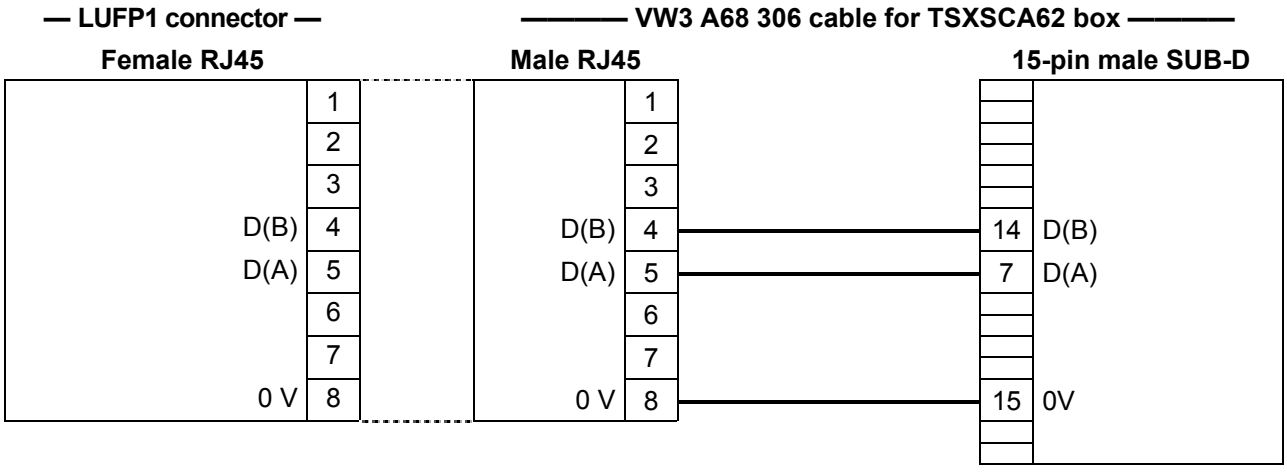
- **"Bus" topology with tap boxes:** This topology is similar to the previous one, except that it uses TSXSCA62 subscriber connectors and / or TSXCA50 subscriber connectors. We recommend using a VW3 A68 306 connection cable and the TSXCSA•00 Modbus cables. Connect the RJ45 connector on the VW3 A68 306 cable to the Modbus connector on the LUFP1 gateway.

  The connections are shown below:



2.5.2. Pin-Outs

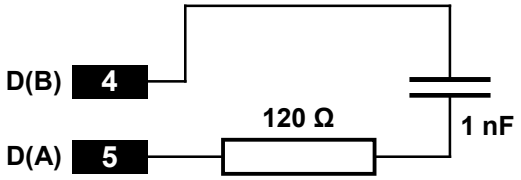
In addition to the pin-out for the connector on the gateway, the one on the VW3 A68 306 cable is also shown below, as it is the only Modbus cable that does not exclusively use RJ45 connections.



2. Hardware Implementation of the LUFP1 Gateway

2.5.3. Wiring Recommendations for the Modbus Network

- Use a shielded cable with 2 pairs of twisted conductors;
- Connect the reference potentials to one another;
- Maximum length of line: 1,000 metres;
- Maximum length of drop line / tap-off: 20 metres;
- Do not connect more than 9 stations to a bus (slaves and one LUFP1 gateway);
- Cable routing: keep the bus away from power cables (at least 30 cm), make crossings at right angles if necessary, and connect the cable shielding to the earth on each unit;
- Adapt the line at both ends using a line terminator (see diagram and VW3 A8 306 RC termination below).



— Line termination recommended at both ends of the line —

— VW3 A8 306 RC line termination —

To make it easier to connect the units using the topologies described in Section 2.5.1 Examples of Modbus Connection Topologies, page 15, various accessories are available in the *Schneider Electric* catalogue:

1) Hubs, drops, taps, and line terminations:

☐ LU9GC03 hub..... (“star” topology)	This passive box has 8 female RJ45 connectors. Each of these connectors can be connected to a Modbus slave, to a Modbus master, to another Modbus hub, or to a line termination.
☐ VW3 A8 306 TF3 drop box..... (“bus” topology with VW3 A8 306 TF3 drop boxes)	This passive box includes a short lead with a male RJ45 connector allowing it to be connected directly to a Modbus slave, without having to use a different cable. It is fitted with 2 female RJ45 connectors for the connection of two Modbus cables of the VW3 A8 306 R•• type.
☐ 2-way TSXSCA62 subscriber connector. (“bus” topology with tap boxes)	This passive box has a printed circuit fitted with screw terminals and allows the connection of 2 subscribers to the bus (2 female 15-pin SUB-D connectors). It includes the line termination when the connector is located at the end. It is fitted with 2 screw terminals for the connection of two double twisted pair Modbus cables.
☐ TSXCA50 tap box..... (“bus” topology with tap boxes)	This passive box allows a Modbus unit to be connected to a screw terminal. It includes the line termination when the connector is located at the end. It is fitted with 2 screw terminals for the connection of two double twisted pair Modbus cables.
☐ VW3 A8 306 RC double termination ..... (all topologies)	Each of these two red passive boxes is a male RJ45 connector 3 cm long containing an RC line termination (see diagram and illustration above). Only the abbreviation “RC” is shown on these boxes.

2. Hardware Implementation of the LUFP1 Gateway

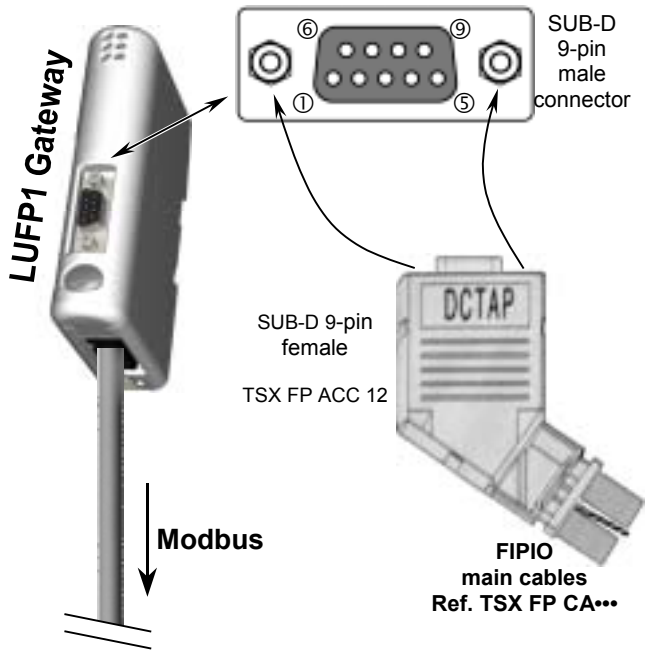
2) Cables:

- VW3 A8 306 R•• Modbus cable ..... Shielded cable with a male RJ45 connector at each end. (“star” topology / “bus” topology with tap boxes)
- VW3 A68 306 Modbus cable..... Shielded cable with a male RJ45 connector and a 15-pin male SUB-D connector. It is used to connect a Modbus subscriber (slave or master) to a TSXSCA62 or TSXCA50 box. (“bus” topology with tap boxes)
- Shielded double twisted pair Modbus cable ..... Bare cable (without connectors) used to make up the main section of the Modbus network. There are three items available: TSXCSA100 (100 m), TSXCSA200 (200 m), and TSXCSA500 (500 m). (“bus” topology with branch boxes)

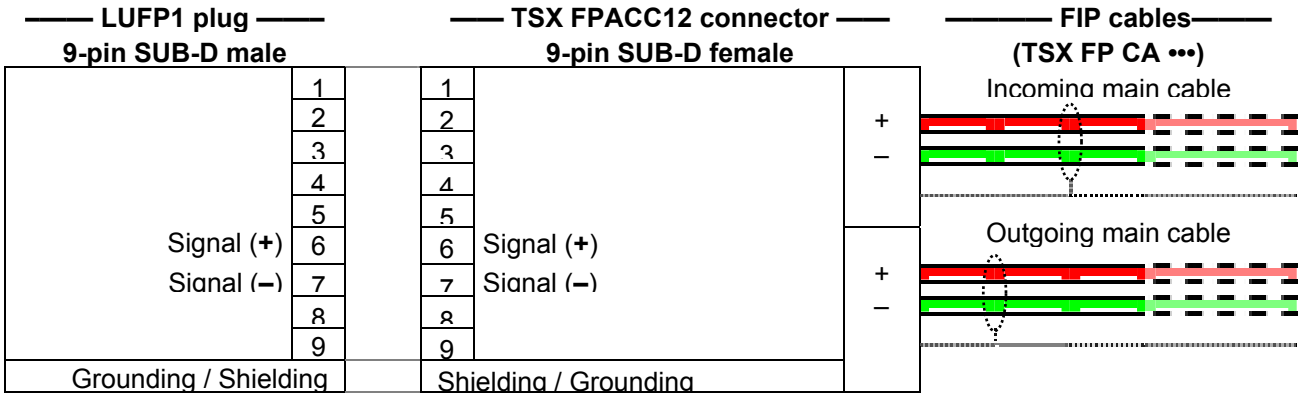
2.6. Connecting the LUFP1 Gateway to the FIPIO Network

Connect the SUB-D 9-pin female plug on the FIPIO connector to the FIPIO plug on the LUFP1 gateway.

The connections are illustrated here:



2.6.1. Pin-Outs



N.B. If the two main cables are replaced by a TSX FP CC ••• tap-off cable, the **red** and **orange** wires correspond to the signal (**+**), while the **green** and **black** wires correspondent to the signal (**–**).

2.6.2. Wiring Recommendations for the FIPIO Network

- Use TSX FP ACC 12 connectors (or TSX FP ACC 2 in the case of TSX 17-20 micro-PLCs) to connect the gateway to the FIPIO network.
- Use a TSX FP CA ••• main FIPIO cable to link up the connectors and a TSX FP CC ••• FIPIO tap-off cable if you need to connect up to a tap box (TSX FP ACC 14, TSX FP ACC 3, TSX FP ACC 4 or TSX FP ACC 10). Both of these shielded cables are available in lengths of 100, 200 or 500 meters.
- Use two line terminations to close off the FIPIO network at both ends of each electrical segment. The TSX FP ACC 7 kit includes two FIP terminations.
- Connect the reference potentials to one another.
- The transmission rate is fixed at 1 Mbit / s.
- Use TSX FP ACC 6 repeaters to connect up two electrical segments on the same FIPIO network. The maximum length of an electrical segment is 1,000 m. Follow this rule:

  [(Number of repeaters ÷ 2) + Sum of lengths (in km)] < 22

  The longest FIPIO network that can be implemented under this rule would be 15 km long and would require 14 repeaters (14 ÷ 2 + 15 = 22).
- Do not connect more than 32 master or slave stations per segment (not counting repeaters).
- Cable routing: keep the bus away from power cables (at least 30 cm), make crossings at right angles if necessary and connect the cable shielding to the earth on each unit.

For more details about connections, see the ***FIPIO Bus / FIPWAY Network Reference Manual*** (ref. TSX DR FIP F for the French version, TSX DR FIP E for the English version, TSX DR FIP G for the German version and TSX DR FIP S for the Spanish version). This manual also contains details on the operating characteristics and implementation of a FIPIO field bus.

In addition, the manual ***Electromagnetic Compatibility of Industrial Field Buses and Networks*** (ref. TSX DGKBL F for the French version, ref. TSX DGKBL E for the English version and ref. TSX DGKBL G for the German version) contains valuable rules and precautionary measures for wiring up a FIPIO field bus.

A number of accessories are available from the *Schneider Electric* catalogue to facilitate the connection of stations on a FIPIO network:

1) Connectors, tap boxes and line terminations:

- □ TSX FP ACC 12 connector .....Isolating connector for equipment with a 9-pin female SUB-D connector. Used for daisy-chain or parallel cabling (see illustration on next page).
- □ TSX FP ACC 2 connector .....Female connector for the TSX FPG 10• coupler (TSX 17-20 micro-PLCs). Used for daisy-chain or parallel cabling.
- □ TSX FP ACC 14 box.....Isolating tap box for bus connections to the main cable.
- □ TSX FP ACC 4 box.....Watertight tap boxes for bus connections to the main cable. They also support one 9-pin female SUB-D connector (for the PCMCIA card cable TSX FP CG 010 / 030).
  TSX FP ACC 10 box
- □ TSX FP ACC 3 box.....Isolated tap box for bus connections to the main cable. Also supports two 9-pin female SUB-D connectors (*as above*).
- □ TSX FP ACC 7 terminations.....Set of two line terminations for adapting the impedance of both ends of an electrical segment. You will need one set per electrical segment.
- □ TSX FP ACC 6 repeater.....Electrical repeater for joining two segments, each with a maximum length of 1,000 m.

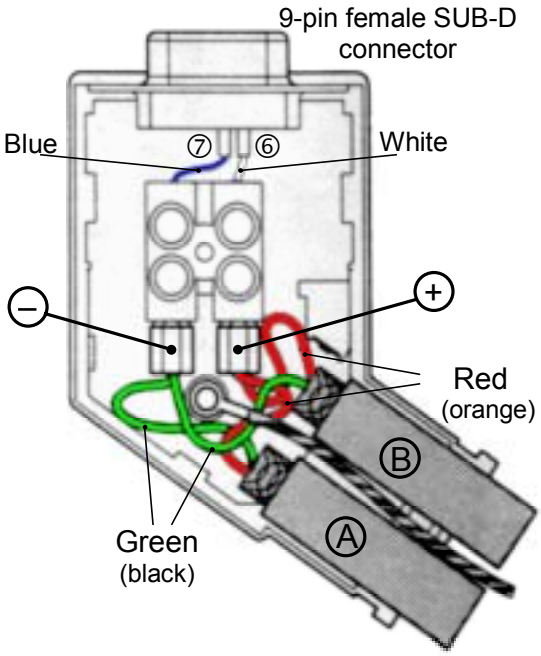
2. Hardware Implementation of the LUFP1 Gateway

2) Cables:

- TSX FP CA••• main cable .....Shielded twisted 150 Ω cable with a single pair of conductors: red (+) and green (–). In lengths of 100, 200 or 500 m. This cable is the main trunk of the FIPIO network and is therefore used to link up connectors and tap boxes.
- TSX FP CC••• tap link cable.....Shielded twisted 150 Ω cable with two pairs of conductors: red or orange (+), and green or black (–). In lengths of 100, 200 or 500 m. This cable is used to connect a tap box to a distant connector.

  N.B. When calculating the length of a FIPIO segment, remember to multiply the length of the cable by two.

TSX FP ACC 12 Connector



The illustration opposite shows the wiring in the TSX FP ACC 12 connector when connecting up to the main cable (two sections of TSX FP CA••• cable).

Replace one of the two cables by one of the two line terminations supplied in the TSX FP ACC 7 kit, if the connector is at the end of an electrical segment.

Replace the two cables (A) and (B) by a TSX FP CC••• cable (in (A)) if the connection point is a derivation rather than a daisy-chain link; if so, the **orange** wire corresponds to the (**+**) signal of the second pair of conductors and the **black** wire corresponds to the (**–**) signal.

ENGLISH

2.6.3. FIPIO Bus Commissioning Procedure

We recommend commissioning the devices in sequence, one after the other. For a detailed description of the initial commissioning of an application on FIPIO, see the ***FIPIO Bus / FIPWAY Network Reference Manual*** referred to on the previous page.

2. Hardware Implementation of the LUFP1 Gateway

2.7. Configuring the FIPIO Communication Features

Configuration should be carried out when the gateway is powered off.

This task is limited to configuring the gateway's FIPIO address, as the transmission rate on the FIPIO network is fixed at 1 Mbit/s.

The two coding wheels used for configuring the gateway's address are hidden behind the gateway cover ⑤ (see illustration in Section 2.2 Overview of the LUFP1 Gateway, page 13). To remove this cover, all you have to do is insert the end of a small flat screwdriver between the top of the hood and the gateway box, and lift it off.

The power supply of the gateway must be turned off before opening the cover.

Once the cover has been removed, make sure that you touch neither the electrical circuits nor the electronic components.

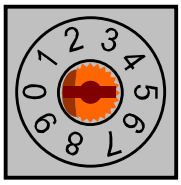
2.7.1. Encoding the Gateway Address

The LUFP1 gateway is identified on the FIPIO bus by its address, ranging from 1 to 99.

On FIPIO, address 0 is reserved for the PLC that manages the bus. Address 63 is reserved for the programming or operation terminal. All the other addresses, up to physical address 127, can be used.

Units

Tens



The gateway's FIPIO address depends on the position of the two coding wheels represented on the left, in their factory-setting positions (default address = 2).

This address is the sum of the decimal values given by the angular positions of the bottom coding wheel (tens) and the top coding wheel (units).

Any change to the gateway's address is taken into account only once it is powered on again.

No two devices on the same FIPIO bus can ever have the same address. Simultaneous, fast (2 Hz) and continuing flashing of LEDs ❶, ❷, ❸ and ❹ means that the device cannot connect to the FIPIO bus because the address is already used by another device.
N.B. This flashing signal also occurs when address 00 is configured using the gateway's coding wheels.

Examples:

Address = 19

Units

Tens

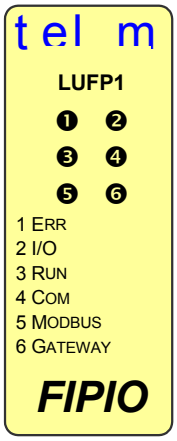
Address = 73

Units

Tens

3. Signalling

The gateway's 6 LEDs and the descriptive label on the removable cover that hides its two coding wheels (for the gateway address) allow you to diagnose the status of the gateway:



LED	LED → Gateway state
❶ ERR	**Off: Gateway on-line (exchanges possible)**
	Flashing (red): Gateway off-line (exchanges impossible) ↳*Check, in PL7 PRO, that a FIPIO station has been declared for the gateway address. Check that the gateway is correctly connected to the FIPIO network.*
❸ RUN	Off: No power, or critical fault
	Flashing green: Duplicate address
	Green: Gateway operational
❺ MODBUS	Off: No power
	Flashing (green): No Modbus communications
	Green: Modbus communications OK with all Modbus slaves
	Red: Loss of communication with one or more Modbus slaves

LED	LED → Gateway state
❷ I / O	**Off: No fault**
	Red: Internal fault ↳*Replace the gateway*
	Flashing (red): Timing failure ↳*Fault mainly due to FIPIO network and / or FIPIO master*
❹ COM	Off: Total absence of communication with bus arbitrator
	Flashing yellow (5 Hz): Exchanges in progress with bus arbitrator
❻ GATEWAY	Off: No power
	Green: Initialisation and configuration of the gateway
	Flashing (green): Gateway in operation
	Flashing (red / green): Configuration absent / invalid ↳*Use AbcConf to load a valid configuration*

ENGLISH

If* LED ❻ *does not flash from red to green and if LEDs ❶, ❷, ❸ and ❹ all flash at the same time at a frequency of 2 Hz, this means that the physical gateway address is already used by another FIPIO device, or that it has been set at 0 with the gateway's coding wheels.

↳*Try to resolve this address conflict by viewing the FIPIO network configuration in PL7 PRO. If necessary, modify the addresses in the configuration. Also check the physical address configured on the gateway using the coding wheels (see Section 2.7.1 Encoding the Gateway Address, page 22).*

N.B. If both identically-addressed devices are reconnected to FIPIO network simultaneously, then the four LEDs mentioned above will not flash. In this case, only LED ❶ will flash, and then only discontinuously and in an irregular manner. This happens, for example, when you disconnect and then reconnect the bus arbitrator.

4. Software Implementation of the Gateway

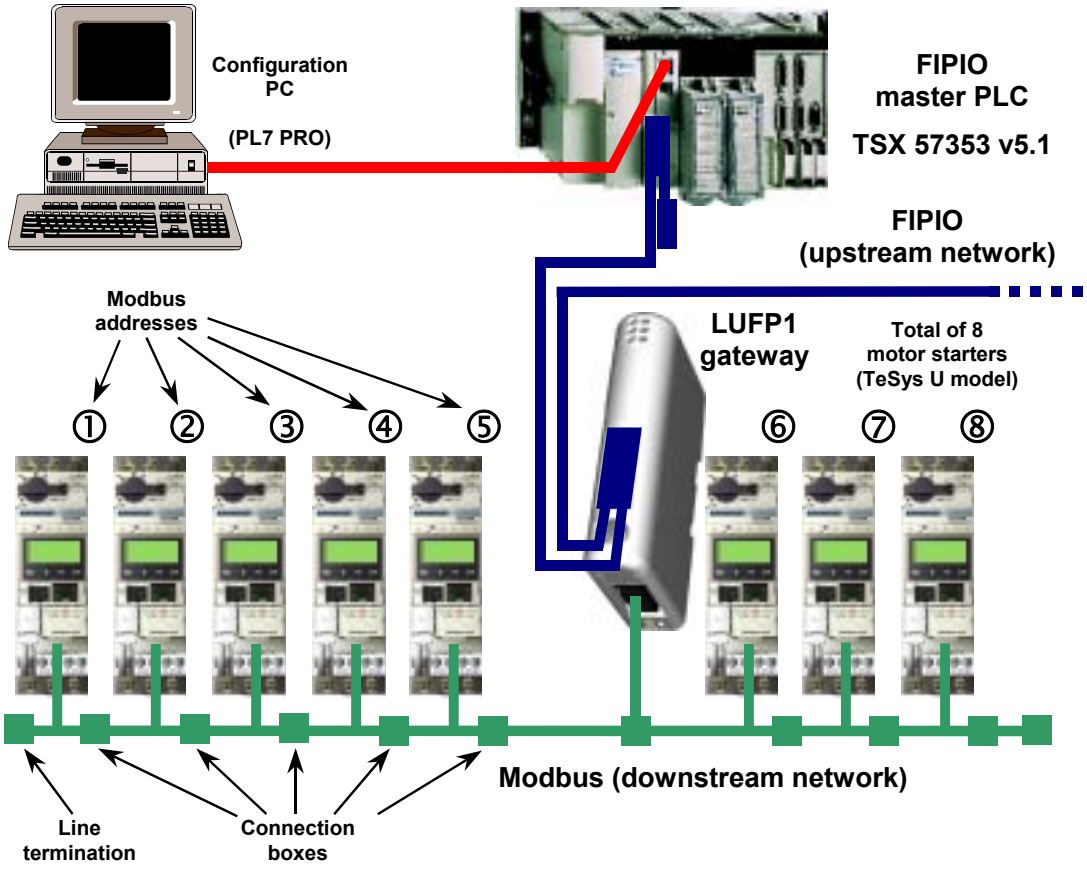
4.1. Introduction

This chapter presents the implementation of the LUFP1 gateway in PL7 PRO, with a *Telemecanique* “Premium” PLC (e.g. PLC TSX 57353 v5.1) acting as master and FIPIO bus arbitrator via the FIPIO link built into its processor card.

The various configuration possibilities for Modbus exchanges on the gateway are not covered in this section, as the aim here is to describe a standard configuration with 8 TeSys U motor starters. This configuration is described with a view to implementing the gateway using PL7 PRO. The full range of gateway configuration possibilities is set out in Appendix B: LUFP1 Gateway Settings (Chapter 9, page 100).

4.1.1. System Architecture

The configuration described in this LUFP1 gateway implementation enables the device to command and control eight TeSys U motor starters:



See Chapter 2 Hardware Implementation of the LUFP1 Gateway, page 13, for details on implementing the hardware for the standard configuration.

ENGLISH

4. Software Implementation of the Gateway

4.1.2. Configuring the Motor Starters

Each motor starter should be configured as follows:

Protocol:	Modbus RTU slave
Modbus address	1 to 8
Bitrate	19,200 bits/s
Data bits	8

Start bits	1
Parity	None
Parity bit	0
Stop bits	1

When using a TeSys U motor starter with a Modbus communication module (LULC031 module), the configuration parameters for the RS485 connection are automatically detected, only the Modbus address of the motor starter needs to be configured.

N.B. The gateway's Modbus slaves cannot be configured with PL7 PRO; the configuration and adjustment of the LUFP1 gateway are limited to the gateway and to the management of generic Modbus exchanges.

4.1.3. Modbus Cycle Time

The LUFP1 gateway's default configuration sets a cycle time of 300 ms on Modbus commands for each of the 8 TeSys U motor starters.

ENGLISH

4.1.4. Managing Degraded Modes

The degraded-mode management of the LUFP1 gateway configuration is described below. In this instance, we are using a Premium PLC with a FIPIO link built into the processor card.

<table>
<tr><th colspan="2" rowspan="2">Observed behaviour</th><th colspan="4">Event</th></tr>
<tr><th>Premium PLC: CPU stop or failure</th><th>Disconnection of the upstream FIPIO network</th><th>Failure of the LUFP1 gateway</th><th>Disconnection of the downstream Modbus network</th></tr>
<tr><td rowspan="3">Outputs</td><td>Reset</td><td colspan="2">Yes</td><td colspan="2" rowspan="2">Depending on the configuration of the TeSys U motor starters (1)</td></tr>
<tr><td>Hold</td><td colspan="2" rowspan="2">——</td></tr>
<tr><td>Stop refreshing</td><td colspan="2">——</td></tr>
<tr><td rowspan="2">Inputs</td><td>Reset</td><td rowspan="2">——</td><td colspan="2">——</td><td>Yes (2)</td></tr>
<tr><td>Hold</td><td colspan="2">Yes (3)</td><td>——</td></tr>
</table>

(1) The desired behaviour with regard to outputs should be directly configured on each TeSys U motor starter.

(2) Use the List of Active Slaves (LAS) Service described in Section 5.2, page 42.

(3) Use the FIPIO Diagnostic Objects (Section 5.4, page 52) to detect the underlying event.

4. Software Implementation of the Gateway

4.2. Gateway Configuration under PL7 PRO

The following sections describe the configuration steps in PL7 PRO that you will need to go through to ensure that the gateway is correctly recognised by the FIPIO master PLC and that it can command and control the 8 motor starters in the standard configuration described earlier.

4.2.1. WorldFIP Mode

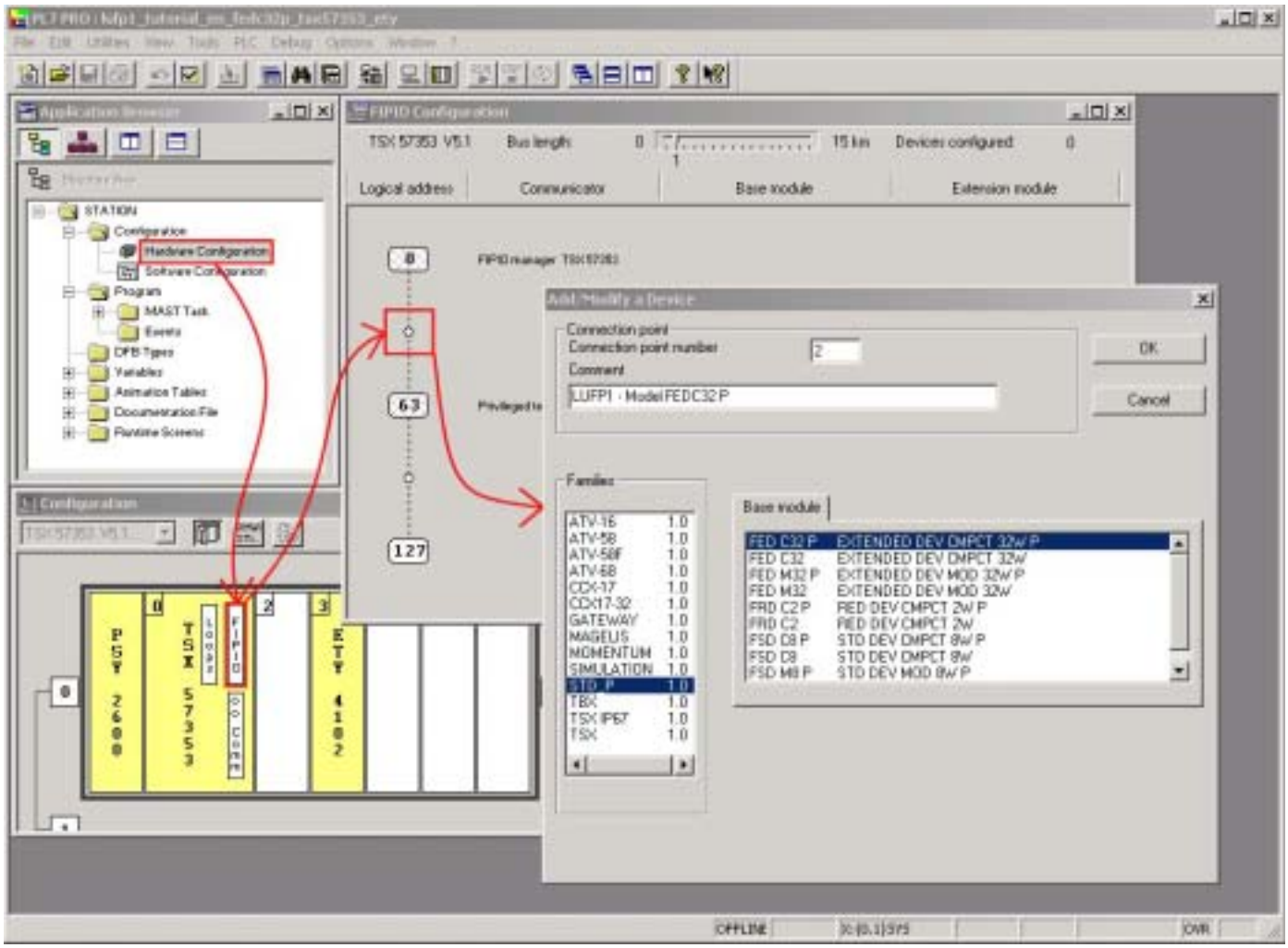
The LUFP1 gateway supports the WorldFIP mode, configured via the FIPIO bus properties in PL7.

4.2.2. Adding the LUFP1 Gateway under PL7 PRO

Open the hardware configuration of your application and edit the FIPIO network configuration by double-clicking Fipio on the PLC processor card (TSX 57353).

Add the FIPIO device corresponding to the LUFP1 gateway by double-clicking the first free connection point. In the window that appears, select the family “STD_P” and the basic module for that family, “FED C32 P”.

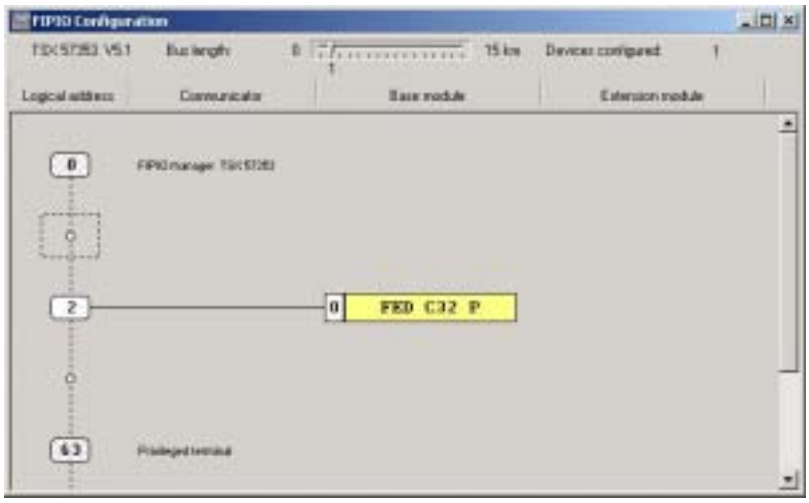
You also need to enter the physical address of the gateway on the FIPIO bus in the field “Connection point number”. Entering a “Comment” is optional, but it does make it easier to pick out the gateway from among the other devices on the same FIPIO bus.



ENGLISH

4. Software Implementation of the Gateway

4.2.3. Configuring and Adjusting the LUFP1 Gateway



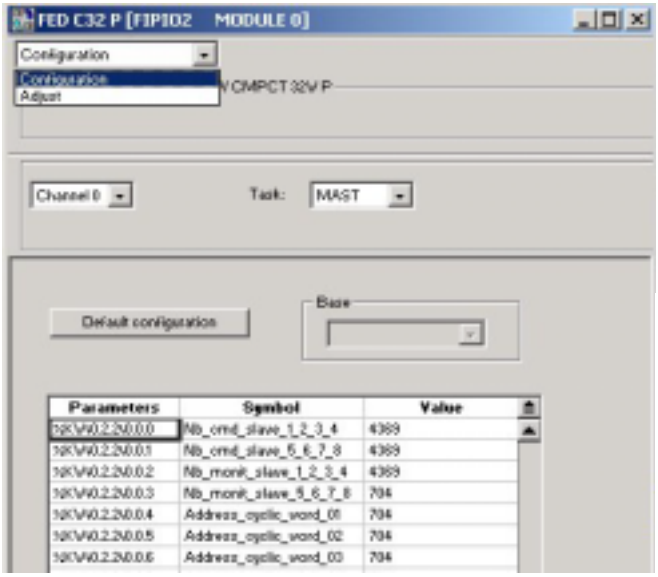
Once the gateway has been added to the FIPIO network (see opposite), double-click the basic module

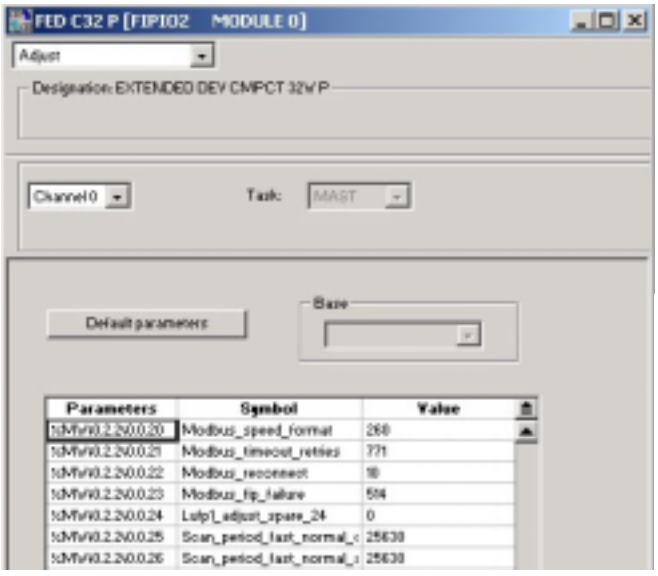
`0 | FED C32 P`

to edit its configuration and adjustment parameters.

In the window that appears, enter the values given on the next two pages. These correspond to the configuration described in Section 4.1.1 System Architecture, page 25.

N.B. Make sure you configure the “Bus Length” correctly (see Section 2.6.2 Wiring Recommendations for the FIPIO Network, page 20).





N.B. The “FED C32 P [FIPIO2 MODULE 0]” windows shown above present “Symbols” which have been added manually using the PL7 PRO variables editor. By default, of course, no symbol is attributed to the gateway’s configuration parameters (%KW) or adjustment parameters (%MW).

All the configuration and adjustment parameters described in the following sections are an integral part of the PLC application. The FIPIO master PLC will transmit these parameters to the gateway implicitly via the FIPIO bus whenever either of the two following events occurs:

- The PLC goes into RUN mode;
- The gateway is disconnected from and reconnected to the FIPIO network.

The gateway configuration possibilities offered by the FED C32 P model are limited to the gateway itself; it cannot be used to configure the Modbus slaves. If you want to configure the latter, you will need to use a different method (local configuration or remote configuration with a software tool such as PowerSuite, for example).

ENGLISH

4. Software Implementation of the Gateway

4.2.4. Configuration Parameter Values, Standard Config

Parameter	Value		Description
%KW\p.2.c\0.0.0	4369	16#1111	Number of periodic command words for slaves n°1 to n°4: Bits 0- 3: Number of periodic command words for slave n°1 = 1 Bits 4- 7: Number of periodic command words for slave n°2 = 1 Bits 8-11: Number of periodic command words for slave n°3 = 1 Bits 12-15: Number of periodic command words for slave n°4 = 1
%KW\p.2.c\0.0.1	4369	16#1111	Number of periodic command words for slaves n°5 to n°8: Bits 0- 3: Number of periodic command words for slave n°5 = 1 Bits 4- 7: Number of periodic command words for slave n°6 = 1 Bits 8-11: Number of periodic command words for slave n°7 = 1 Bits 12-15: Number of periodic command words for slave n°8 = 1
%KW\p.2.c\0.0.2	4369	16#1111	Number of periodic control words for slaves n°1 to n°4: Bits 0- 3: Number of periodic control words for slave n°1 = 1 Bits 4- 7: Number of periodic control words for slave n°2 = 1 Bits 8-11: Number of periodic control words for slave n°3 = 1 Bits 12-15: Number of periodic control words for slave n°4 = 1
%KW\p.2.c\0.0.3	4369	16#1111	Number of periodic control words for slaves n°5 to n°8: Bits 0- 3: Number of periodic control words for slave n°5 = 1 Bits 4- 7: Number of periodic control words for slave n°6 = 1 Bits 8-11: Number of periodic control words for slave n°7 = 1 Bits 12-15: Number of periodic control words for slave n°8 = 1
%KW\p.2.c\0.0.4	704	16#02C0	Address of periodic word n° 1: Address of the unique periodic command word for slave n°1 (704 → command word for a TeSys U motor starter)
%KW\p.2.c\0.0.5	704	16#02C0	Address of periodic word n° 2: *As above*, but for slave n°2
%KW\p.2.c\0.0.6	704	16#02C0	Address of periodic word n° 3: *As above*, but for slave n°3
%KW\p.2.c\0.0.7	704	16#02C0	Address of periodic word n° 4: *As above*, but for slave n°4
%KW\p.2.c\0.0.8	704	16#02C0	Address of periodic word n° 5: *As above*, but for slave n°5
%KW\p.2.c\0.0.9	704	16#02C0	Address of periodic word n° 6: *As above*, but for slave n°6
%KW\p.2.c\0.0.10	704	16#02C0	Address of periodic word n° 7: *As above*, but for slave n°7
%KW\p.2.c\0.0.11	704	16#02C0	Address of periodic word n° 8: *As above*, but for slave n°8
%KW\p.2.c\0.0.12	455	16#01C7	Address of periodic word n° 9: Address of the unique periodic control word for slave n°1 (455 → status word for a TeSys U motor starter)
%KW\p.2.c\0.0.13	455	16#01C7	Address of periodic word n°10: *As above*, but for slave n°2
%KW\p.2.c\0.0.14	455	16#01C7	Address of periodic word n°11: *As above*, but for slave n°3
%KW\p.2.c\0.0.15	455	16#01C7	Address of periodic word n°12: *As above*, but for slave n°4
%KW\p.2.c\0.0.16	455	16#01C7	Address of periodic word n°13: *As above*, but for slave n°5
%KW\p.2.c\0.0.17	455	16#01C7	Address of periodic word n°14: *As above*, but for slave n°6
%KW\p.2.c\0.0.18	455	16#01C7	Address of periodic word n°15: *As above*, but for slave n°7
%KW\p.2.c\0.0.19	455	16#01C7	Address of periodic word n°16: *As above*, but for slave n°8
%KW\p.2.c\0.0.20	0	16#0000	Address of periodic word n°17: Not used (bits at 0)
……	……	……	……
%KW\p.2.c\0.0.29	0	16#0000	Address of periodic word n°26: Not used (bits at 0)

Syntax: p = processor position (0 or 1);
c = connection point number (gateway address).

For modifications to these adjustment parameters, see Appendix B: LUFP1 Gateway Settings (Chapter 9, page 100).

However, during the Software Implementation of the Gateway, you will probably find it useful to know how to delete one or more TeSys U motor starters. This operation is therefore described in Section 4.2.6 Deleting one or more TeSys U Motor Starters from the Configuration, page 31.

ENGLISH

4. Software Implementation of the Gateway

4.2.5. Adjustment Parameter Values, Standard Configuration

Parameter	Value		Description
%MW\p.2.c\0.0.20	260	16#0104	Bits 0- 7: Modbus speed = 19,200 bits/s (LSB byte = 4) Bit 8: Modbus format = 8 data bits (bit at 1) Bit 9: Modbus format = 1 stop bit (bit at 0) Bits 10-11: Modbus format = No parity (bits at 0) Bits 12-15: Not used (bits at 0)
%MW\p.2.c\0.0.21	771	16#0303	Bits 0- 7: Modbus response timeout = 300 ms (3 * 100 ms) Bits 8-15: Number of frame retransmissions over the Modbus network = 3
%MW\p.2.c\0.0.22	10	16#000A	Bits 0- 7: Time taken by gateway to reconnect Modbus slave following a break in communication = 10 s (10 × 1 s) Bits 8-15: Not used (bits at 0)
%MW\p.2.c\0.0.23	514	16#0202	Bits 0- 7: Behaviour on losing FIPIO network = Reset periodic command word values sent to the Modbus slaves (LSB byte = 2) Bits 8-15: Behaviour on losing Modbus network = Reset periodic control word values sent by the Modbus slaves (LSB byte = 2)
%MW\p.2.c\0.0.24	0	16#0000	Not used (bits at 0)
%MW\p.2.c\0.0.25	25 630	16#641E	Bits 0- 7: Cycle time of fast periodic commands = 300 ms (30 × 10 ms) Bits 8-15: Cycle time of normal periodic commands = 1 s (100 × 10 ms)
%MW\p.2.c\0.0.26	25 630	16#641E	Bits 0- 7: Cycle time of fast periodic controls = 300 ms (30 × 10 ms) Bits 8-15: Cycle time of normal periodic controls = 1 s (100 × 10 ms)
%MW\p.2.c\0.0.27	0	16#0000	Cycle time settings for periodic commands n°1 to 16: Bit 0: Periodic command n°1 = fast (bit at 0) Bits 1 to 7: *As above*, but for commands n°2 to 8, in that order Bits 8 to 15: Not used (bits at 0)
%MW\p.2.c\0.0.28	0	16#0000	Cycle time settings for periodic commands n°17 to 26: Not used
%MW\p.2.c\0.0.29	0	16#0000	Cycle time settings for periodic controls n°1 to 16: Bit 0: Periodic control n°1 = fast (bit at 0) Bits 1 to 7: *As above*, but for controls n°2 to 8, in that order Bits 8 to 15: Not used (bits at 0)
%MW\p.2.c\0.0.30	0	16#0000	Cycle time settings for periodic controls n°17 to 26: Not used
%MW\p.2.c\0.0.31	513	16#0201	**Modbus addresses:** Slave n°1 (bits 0-7) = 1 — Slave n°2 (bits 8-15) = 2
%MW\p.2.c\0.0.32	1 027	16#0403	**Modbus addresses:** Slave n°3 (bits 0-7) = 3 — Slave n°4 (bits 8-15) = 4
%MW\p.2.c\0.0.33	1 541	16#0605	**Modbus addresses:** Slave n°5 (bits 0-7) = 5 — Slave n°6 (bits 8-15) = 6
%MW\p.2.c\0.0.34	2 055	16#0807	**Modbus addresses:** Slave n°7 (bits 0-7) = 7 — Slave n°8 (bits 8-15) = 8
%MW\p.2.c\0.0.35	0	16#0000	Not used (bits at 0)
.....	.....	.....	.....
%MW\p.2.c\0.0.49	0	16#0000	Not used (bits at 0)

Syntax: p = processor position (0 or 1);
c = connection point number (gateway address).

As you can see from parameters %MW\p.2.c\0.0.31 to %MW\p.2.c\0.0.34, care must be taken not to confuse the order of configuration of the Modbus slaves with their Modbus addresses.

The configuration shown here could cause exactly that confusion in that the order numbers of the Modbus slaves (1 to 8) are identical to the addresses of the corresponding TeSys U motor starters (Modbus addresses 1 to 8).

For modifications to these adjustment parameters, see Appendix B: LUFP1 Gateway Settings (Chapter 9, page 100).

However, during the Software Implementation of the Gateway, you will probably find it useful to know how to delete one or more TeSys U motor starters. This operation is therefore described in Section 4.2.6 Deleting one or more TeSys U Motor Starters from the Configuration, page 31.

ENGLISH

4.2.6. Deleting one or more TeSys U Motor Starters from the Configuration

We will just look at one modification operation on the standard 8 TeSys U motor starter configuration: the deletion of one or more motor starters. The 8 motor starters were configured to represent the typical *maximal* configuration (a maximum of 8 Modbus slaves).

It is best to delete any slaves absent from your configuration in order to avoid any degradation in performance that might be caused by the physical absence of motor starters declared in the LUFP1 gateway configuration.

Make the following changes to the Configuration Parameter Values, Standard Config (Section 4.2.4, page 29) and to the Adjustment Parameter Values, Standard Configuration (Section 4.2.5, page 30):

① For each absent TeSys U motor starter, reset to zero the number of periodic command words (%KW\p.2.c\0.0.0 and %KW\p.2.c\0.0.1) *and* the number of periodic control words (%KW\p.2.c\0.0.2 and %KW\p.2.c\0.0.3) on a single slave, starting with slave n°8 and working back towards slave n°1.

② For each absent TeSys U motor starter, *two* of the addresses of periodic words n°1 to 16 (%KW\p.2.c\0.0.4 to %KW\p.2.c\0.0.19) will become “Not used” parameters, starting with n°16 and working back towards n°1. These two words comprise a command word / control word pair, which means that you have to delete the last command word, shift the control words along by one parameter, and then delete the last control word.

③ Check the mappings between Modbus slaves n°1 to 8 and the addresses of the TeSys U motor starters in your configuration (%MW\p.2.c\0.0.31 to %MW\p.2.c\0.0.34). For each absent motor starter, one of the slaves will take the value 16#00 as its Modbus address, starting with slave n°8 and working back to n°1.

Example: Deletion of the motor starters corresponding to Modbus addresses 1, 3, 4 and 8. The starters at Modbus addresses 2, 5, 6 and 7 will remain present. The new configuration and settings for the gateway are as follows:

Parameter	Value		Description
%KW\p.2.c\0.0.0	4369	16#1111	Periodic command word for each slave from n°1 to n°4
%KW\p.2.c\0.0.1	0	16#0000	Not used
%KW\p.2.c\0.0.2	4369	16#1111	Periodic control word for each slave from n°1 to n°4
%KW\p.2.c\0.0.3	0	16#0000	Not used
%KW\p.2.c\0.0.4	704	16#02C0	Address of periodic word n° 1: Address of the command word for slave n°1
%KW\p.2.c\0.0.5	704	16#02C0	Address of periodic word n° 2: *As above*, but for slave n°2
%KW\p.2.c\0.0.6	704	16#02C0	Address of periodic word n° 3: *As above*, but for slave n°3
%KW\p.2.c\0.0.7	704	16#02C0	Address of periodic word n° 4: *As above*, but for slave n°4
%KW\p.2.c\0.0.8	455	16#01C7	Address of periodic word n° 5: Address of the control word for slave n°1
%KW\p.2.c\0.0.9	455	16#01C7	Address of periodic word n° 6: *As above*, but for slave n°2
%KW\p.2.c\0.0.10	455	16#01C7	Address of periodic word n° 7: *As above*, but for slave n°3
%KW\p.2.c\0.0.11	455	16#01C7	Address of periodic word n° 8: *As above*, but for slave n°4
%KW\p.2.c\0.0.12	0	16#0000	Address of periodic word n° 9: Not used
……	……	……	……
%KW\p.2.c\0.0.29	0	16#0000	Address of periodic word n°26: Not used

4. Software Implementation of the Gateway

Parameter	Value		Description
%MW\p.2.c\0.0.20	260	16#0104	Modbus format and speed
%MW\p.2.c\0.0.21	771	16#0303	Modbus timeout and number of retransmissions
%MW\p.2.c\0.0.22	10	16#000A	Reconnection time for a Modbus slave
%MW\p.2.c\0.0.23	514	16#0202	Behaviour on losing FIPIO network and Modbus network
%MW\p.2.c\0.0.24	0	16#0000	Not used
%MW\p.2.c\0.0.25	25 630	16#641E	Cycle time of fast / normal periodic commands
%MW\p.2.c\0.0.26	25 630	16#641E	Cycle time of fast / normal fast controls
%MW\p.2.c\0.0.27	0	16#0000	Bits 0 to 3: Periodic commands n°1 to 4 = fast (bits at 0) Bits 4 to 15: Not used
%MW\p.2.c\0.0.28	0	16#0000	Not used
%MW\p.2.c\0.0.29	0	16#0000	Bits 0 to 3: Periodic controls n°1 to 4 = fast (bits at 0) Bits 4 to 15: Not used
%MW\p.2.c\0.0.30	0	16#0000	Not used
%MW\p.2.c\0.0.31	1 282	16#0502	**Modbus addresses:** Slave n°1 (bits 0-7) = 2 — Slave n°2 (bits 8-15) = 5
%MW\p.2.c\0.0.32	1 798	16#0706	**Modbus addresses:** Slave n°3 (bits 0-7) = 6 — Slave n°4 (bits 8-15) = 7
%MW\p.2.c\0.0.33	0	16#0000	**Modbus addresses:** Slaves n°5 and 6 absent
%MW\p.2.c\0.0.34	0	16#0000	**Modbus addresses:** Slaves n°7 and 8 absent
%MW\p.2.c\0.0.35	0	16#0000	Not used
……	……	……	……
%MW\p.2.c\0.0.49	0	16#0000	Not used

4.2.7. Configuring Gateway Inputs / Outputs

ENGLISH

The settings described in the previous sections enable the gateway to establish a Modbus scanner made up of periodic write commands (“commands”) and read instructions (“controls”), which are sent out to the Modbus slaves.

Each control word corresponds to a Modbus periodic read instruction, and each command word to a Modbus periodic write command. Each words also corresponds to a gateway input or output. All of these inputs and outputs are defined by the gateway by means of its configuration parameters (%KW): the periodic control words correspond to the inputs, and the periodic command words to the outputs.

The gateway establishes the following correspondences between its inputs / outputs and the values of its periodic command / control words:

Inputs	Periodic control words
%IW\p.2.c\0.0	First control word from slave n°1
:	……
:	Last control word from slave n°1
:	First control word from slave n°2
:	……
:	Last control word from slave n°2
:	:::
:	:::
:	First control word from slave n°8
:	……
:	Last control word from slave n°8
:	:::
%IW\p.2.c\0.0.25	:::

Outputs	Periodic command words
%QW\p.2.c\0.0	First command word for slave n°1
:	……
:	Last command word for slave n°1
:	First command word for slave n°2
:	……
:	Last command word for slave n°2
:	:::
:	:::
:	First command word for slave n°8
:	……
:	Last command word for slave n°8
:	:::
%QW\p.2.c\0.0.25	:::

4. Software Implementation of the Gateway

For the standard configuration presented here, the mapping between the gateway inputs and the PLC inputs is as follows:

Service	PLC input	Description	
		Bit 15.....Bit 8	**Bit 7.....Bit 0**
Periodic communications — Control (monitoring) of TeSys U motor starters	**%IW\p.2.c\0.0**	Value of status register for motor starter ①	
	%IW\p.2.c\0.0.1	Value of status register for motor starter ②	
	%IW\p.2.c\0.0.2	Value of status register for motor starter ③	
	%IW\p.2.c\0.0.3	Value of status register for motor starter ④	
	%IW\p.2.c\0.0.4	Value of status register for motor starter ⑤	
	%IW\p.2.c\0.0.5	Value of status register for motor starter ⑥	
	%IW\p.2.c\0.0.6	Value of status register for motor starter ⑦	
	%IW\p.2.c\0.0.7	Value of status register for motor starter ⑧	
Periodic communications — Free locations	**%IW\p.2.c\0.0.8 ::::::::: %IW\p.2.c\0.0.25**	Not used (18 words)	
—————	**%IW\p.2.c\0.0.26**	Reserved (1 word)	
List of active slaves (LAS) service	**%IW\p.2.c\0.0.27**	List of active Modbus slaves	
Aperiodic communications — Indexed periodic variables (PKW) service – RESPONSE	**%IW\p.2.c\0.0.28**	PKE – Address of the written / read datum	
	%IW\p.2.c\0.0.29	DN – Address of the responding slave	R/W/N – Read / Write OK / Error
	%IW\p.2.c\0.0.30	PWE – Value written / read (1st word)	
	%IW\p.2.c\0.0.31	PWE – Value written / read (2nd word)	

Likewise, the PLC outputs correspond to the gateway outputs as follows:

Service	PLC output	Description	
		Bit 15.....Bit 8	**Bit 7.....Bit 0**
Periodic communications — Command of TeSys U motor starters	**%QW\p.2.c\0.0**	Value of command register for motor starter ①	
	%QW\p.2.c\0.0.1	Value of command register for motor starter ②	
	%QW\p.2.c\0.0.2	Value of command register for motor starter ③	
	%QW\p.2.c\0.0.3	Value of command register for motor starter ④	
	%QW\p.2.c\0.0.4	Value of command register for motor starter ⑤	
	%QW\p.2.c\0.0.5	Value of command register for motor starter ⑥	
	%QW\p.2.c\0.0.6	Value of command register for motor starter ⑦	
	%QW\p.2.c\0.0.7	Value of command register for motor starter ⑧	
Periodic communications — Free locations	**%QW\p.2.c\0.0.8 ::::::::: %QW\p.2.c\0.0.25**	Not used (18 words)	
—————	**%QW\p.2.c\0.0.26 %QW\p.2.c\0.0.27**	Reserved (2 words)	
Aperiodic communications — Indexed periodic variables (PKW) service – COMMAND	**%QW\p.2.c\0.0.28**	PKE – Address of the datum to be read / written	
	%QW\p.2.c\0.0.29	DN – Address of the queried slave	R/W – Read / write command
	%QW\p.2.c\0.0.30	PWE – Value to be written (1st word)	
	%QW\p.2.c\0.0.31	PWE – Value to be written (2nd word)	

ENGLISH

4.2.8. Description of Services Assigned to Gateway I / O

On the FIPIO network, all the inputs / outputs that correspond to the gateway are periodic; the same is not true of the Modbus network. There are therefore two types of service: periodic services and aperiodic services.

All of these services correspond to FIPIO objects. For a more detailed description, see Chapter 5 FIPIO Objects Available for Programming, page 38.

Periodic communications (inputs): For the configuration described here, the value of each of the first 8 gateway input words corresponds to the value of the status register of a TeSys U motor starter (register located at address 455).

Periodic communications (outputs): Likewise, the value of each of the first 8 gateway output words corresponds to a value to be sent to the command register of a TeSys U motor starter (register located at address 704).

The FIPIO objects corresponding to the gateway's periodic Modbus communications are described in Section 5.1 Periodic Command and Control Words, page 39.

See also Section 11.3 Command and Control of the 8 TeSys U Motor Starters, page 115, for an example of how these "periodic communications" services are used with the type of configuration adopted in this gateway implementation (8 TeSys Us).

If you delete any motor starters from the configuration, you will need to adjust the two tables on the previous page accordingly.

ENGLISH

Aperiodic communications: The aperiodic communications service operates in the same way as the "indexed periodic variables", or PKW, service found on a number of other *Schneider Electric* products, such as some ATV drives. Hence the service continues to use the name "PKW".

The FIPIO objects corresponding to the gateway's aperiodic Modbus communications are described in Section 5.3 Indexed Periodic Variables (PKW) Service, page 43.

See also Section 11.4 Using the Indexed Periodic Variables (PKW) Service, page 117, for an example of the simplified use of the gateway's "aperiodic communications" service.

List of active slaves (LAS): This service is specific to the gateway, and does not generate any communications over the Modbus network. That is why it is named differently from the other gateway inputs / outputs (periodic or aperiodic communications). Each bit (from 0 to 7) of the input word indicates the absence or presence of a Modbus slave (slaves n°1 to 8, in that order).

The only FIPIO object corresponding to the list of active slaves service is described in Section 5.2 List of Active Slaves (LAS) Service, page 42.

4. Software Implementation of the Gateway

4.2.9. Validating and Saving the FIPIO Network Configuration

➔ Confirm the FED C32 P profile settings and configuration by clicking the ☑ button.

➔ Close the FED C32 P profile settings and configuration window.

➔ In the FIPIO network configuration window, click ☑ again.

➔ Close this window, and the PLC hardware configuration window.

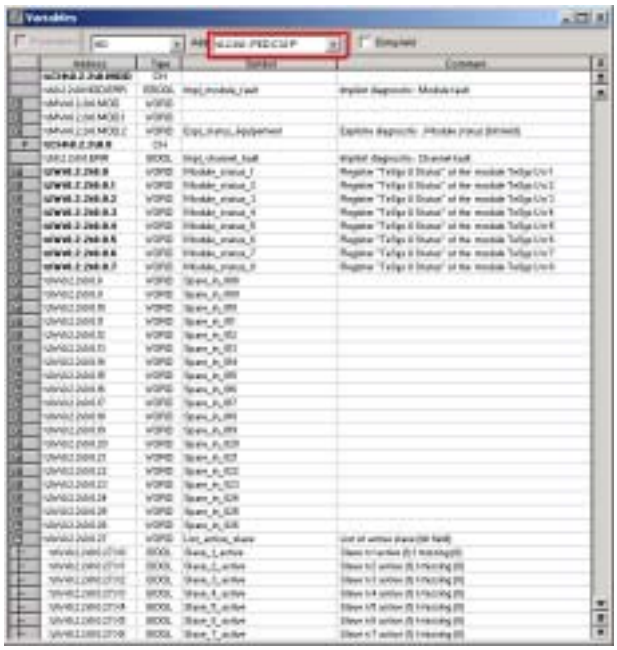
➔ Save the PL7 PRO application by giving it a name.

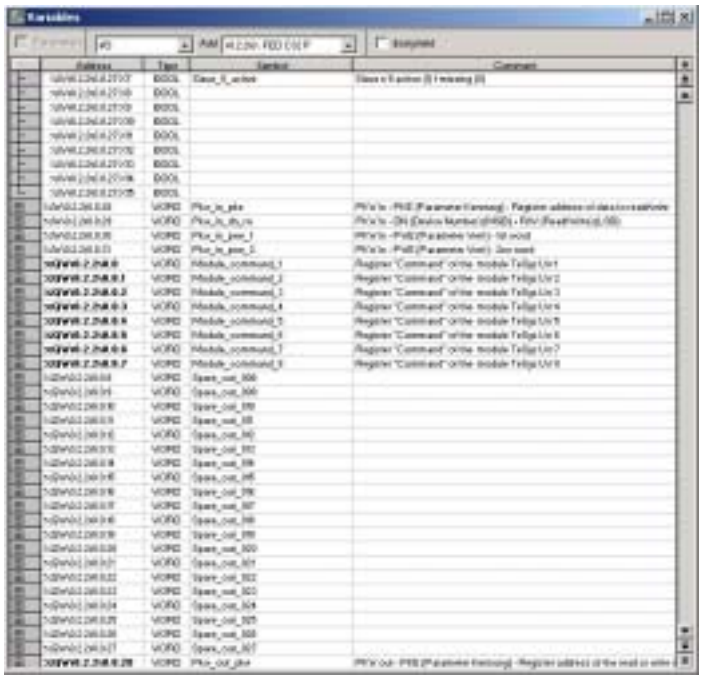
4.2.10. Assigning Symbols to the Gateway's FIPIO Objects

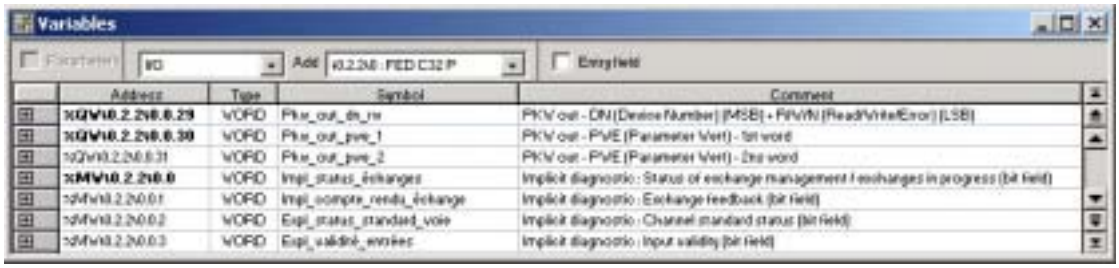
It is advisable to assign symbols to the gateway's inputs and outputs: this makes it easier to understand the PLC application that will be managing them. Edit I / O variables using PL7 PRO, and select the address "\0.2.2\0: FED C32 P" if, for example, the gateway is located at address 2.

You can also assign symbols to the configuration and adjustment parameters described in the previous sections, as well as to the gateway's FIPIO diagnostic objects, but this is not as useful as it is for the inputs and outputs.

Once defined, these symbols are used in the various FED C32 P profile configuration and monitoring windows.







ENGLISH

4.2.11. Checking the Operational Status of the Gateway

After validating any changes made, you must send the new configuration and adjustment parameters for profile FED C32 P to the corresponding gateway by downloading the PL7 PRO application to the PLC. Then check that the gateway has been duly recognised on FIPIO by the PLC, by viewing its status in the FIPIO network configuration window.

➔ Transfer the application from the PC to the PLC by selecting the "Transfer program…" command in the "AP" menu (or by clicking the button) and selecting "PC -> PLC".

➔ Switch from OFFLINE to ONLINE mode by selecting "Connect" in the "AP" menu (or by clicking).

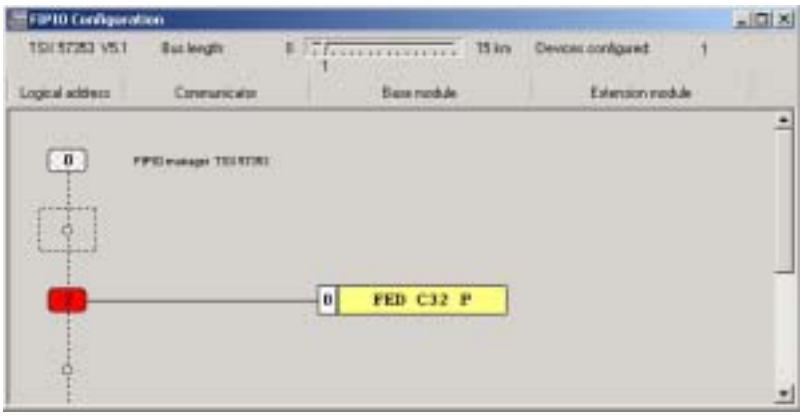
➔ Initialise and start the PLC application using the "Init…" and "Run…" commands in the "AP" menu.

➔ Open the PLC hardware configuration and the FIPIO configuration and check that the gateway connection point is valid (coloured white).

In the example shown here, the connection point is signalling an error (and is therefore in red). If this happens, open the "debug" window by double-clicking the basic module 0 | FED C32 P.

In this window, click on the two red "DIAG…" buttons.

The table below sets out the procedure to be followed in the two most likely instances:



Module diagnosis			Channel diagnosis			Causes and corrective measures
Internal faults	External faults	Other faults	Internal faults	External faults	Other faults	
—	Functional fault	—	—	—	Application fault	**Incorrect configuration / adjustment parameters** ↳ *Switch to LOCAL mode, modify the gateway's parameter values, save the changes, and reload the application onto the FIPIO master PLC.*
—	Functional fault	—	—	—	Hardware configuration fault	**Gateway configured in FED C32** ↳*Update the internal gateway configuration using the AbcConf configuration tool (see Section 7.4.1* Transferring the Internal Configuration (Profile FED C32 P), *page 64).*

Neither of these configuration errors will be signalled by the gateway (its LEDs will indicate normal operation).

It is therefore ***essential*** that you perform the foregoing checks.

4. Software Implementation of the Gateway

4.2.12. Modifying the Gateway Parameters in ONLINE Mode

The modification of one or more configuration and / or adjustment parameters in ONLINE mode will momentarily bring the gateway to a halt (for complete reconfiguration of the Modbus scanner). This interrupts the periodic Modbus communications, but for such a brief period of time that the Modbus slaves are unlikely to detect it, even if they have very short communication timeouts.

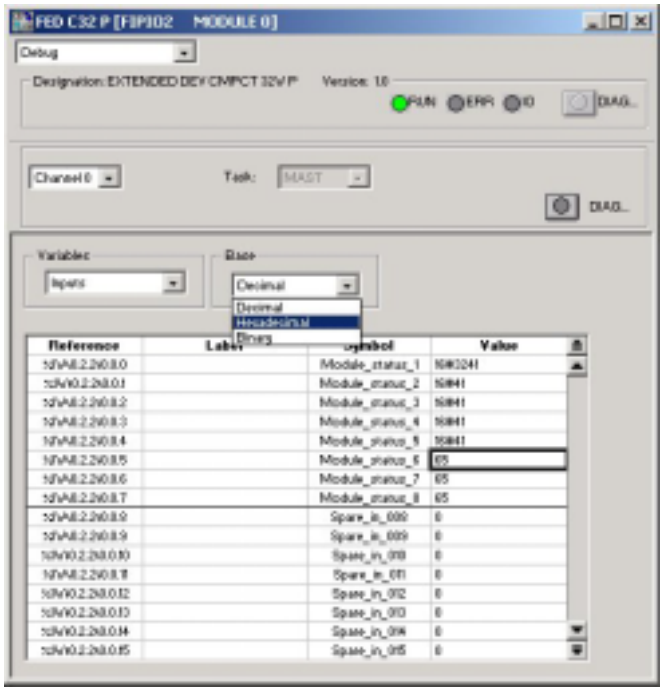
4.2.13. Debugging and Using the Gateway Configuration

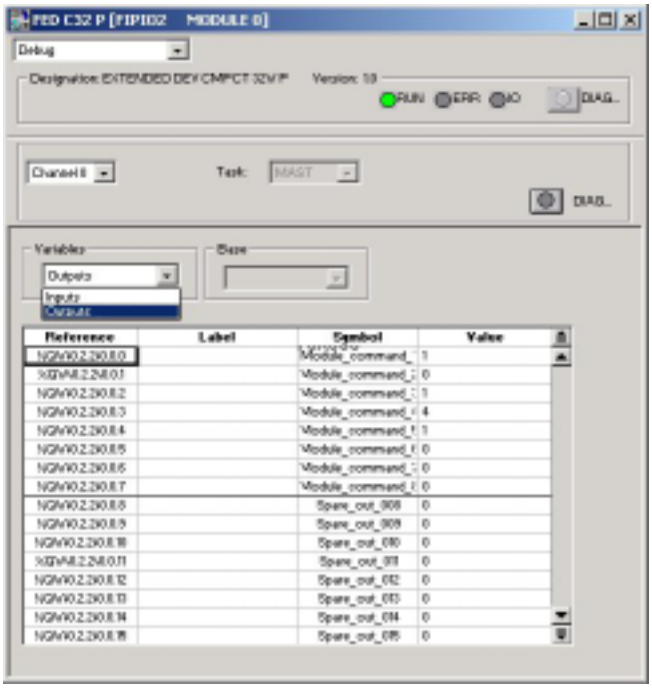
While in ONLINE mode, open the gateway debug window by double-clicking the basic module **0 | FED C32 P**.

This window displays the gateway diagnostics (see previous page), and the values of the gateway inputs *or* outputs (selected using the scroll bar in the “Variables” pane).

The display mode for the selected value can be modified (in the “Base” pane) to make it easier to read if, for example, it consists of two independent bytes or a register of 16 independent bits.

The values of the outputs can also be modified, so that you can check —in the absence of a PLC program that updates the values periodically—that the periodic commands have been transmitted to the gateway, first of all, and subsequently to the TeSys U motor starters.





ENGLISH

4.2.14. Developing a FIPIO Application

The FIPIO master PLC used in the example is a TSX 57353 v5.1, marketed by *Telemecanique*. A sample PLC application, developed under PL7 PRO, is presented in Chapter 11 Appendix D: Sample Use under PL7 PRO, page 114. This example uses the PLC, the gateway and the 8 TeSys U motor starters shown in the Software Implementation of the Gateway.

5. FIPIO Objects Available for Programming

This section presents all the language objects associated with standard profiles FED C32 and FED C32 P (depending on the chosen configuration method—PL7 PRO or AbcConf) for FIPIO communication with the LUFP1 gateway.

All the FIPIO objects described in the following sections (*e.g.* output word %QW\p.2.c\0.0.7) obey the following syntax:

p = processor position (0 or 1);
c = connection point number (gateway address).

The table below sets out all the FIPIO objects available for use with the gateway. They are then described in detail in the rest of the section.

FIPIO Object	Designation	Format	Access (1)
%I\p.2.c\0.0.ERR	Channel fault (if bit at 1)	1 bit	IMPLICIT read
%I\p.2.c\0.MOD.ERR	Module fault (if bit at 1)	1 bit	
%IW\p.2.c\0.0 → \0.0.25	Periodic input variables (control of Modbus slaves)	26 words	IMPLICIT read
%IW\p.2.c\0.0.26	Reserved (word = 16#0000)	1 word	
%IW\p.2.c\0.0.27	List of active slaves (LAS) service	1 word	
%IW\p.2.c\0.0.28 → \0.0.31	Indexed periodic variables (PKW) service: Response	4 words	
%QW\p.2.c\0.0 → \0.0.25	Periodic output variables (command of Modbus slaves)	26 words	IMPLICIT write
%QW\p.2.c\0.0.26 → \0.0.27	Reserved (words not used by gateway)	2 words	
%QW\p.2.c\0.0.28 → \0.0.31	Indexed periodic variables (PKW) service: Command	4 words	
%MW\p.2.c\0.0	Explicit exchanges: Exchange in progress	1 word	IMPLICIT read
%MW\p.2.c\0.0.1	Explicit exchanges: Report on last exchange	1 word	
%MW\p.2.c\0.MOD.2	Module status (gateway diagnosis)	1 word	EXPLICIT read
%MW\p.2.c\0.0.2	Channel status (gateway and communication diagnosis)	1 word	
%MW\p.2.c\0.0.3	Reserved (word = 16#0000)	1 word	EXPLICIT read
%MW\p.2.c\0.0.4 → \0.0.11	Reserved (words = 16#FFFF)	8 words	
%MW\p.2.c\0.0.12 → \0.0.19	Reserved (words = 16#0000)	8 words	EXPLICIT write
%MW\p.2.c\0.0.20 → \0.0.49	Adjustment parameters for profile FED C32 P (2)	30 words	EXPLICIT write
%KW\p.2.c\0.0 → \0.0.29	Configuration parameters for profile FED C32 P (2)	30 words	
%SW128 → %SW135	Faulty FIPIO connection point (if corresponding bit at 0)	8 words	System

(1) IMPLICIT exchanges are performed by the PLC *without any additional programming*; these are the periodic FIPIO exchanges. EXPLICIT exchanges call for the *programming of communication functions* like READ_, PARAMREAD_STS, etc.; these are aperiodic FIPIO exchanges for passing diagnostic variables, downloading complete configurations, and so on.

(2) If the gateway is configured using AbcConf (see Chapter 6 Advanced Implementation of the Gateway, page 56), the profile FED C32 must be used. The configuration and adjustment parameters then become internal to the gateway and no longer need to be transmitted by the FIPIO PLC, and the associated %MW and %KW objects cease to exist.

5. FIPIO Objects Available for Programming

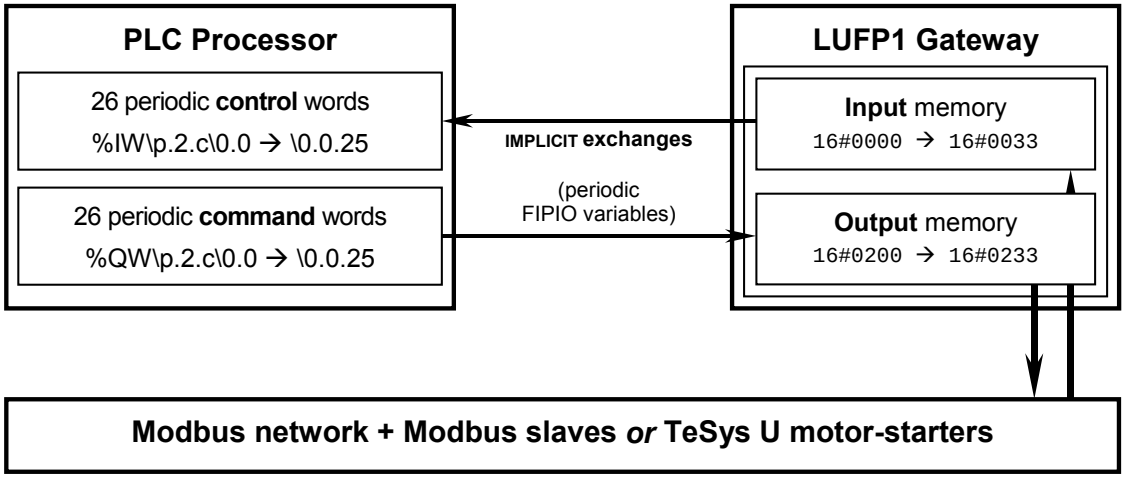
5.1. Periodic Command and Control Words

Unlike the other FIPIO objects for the LUFP1 gateway, the periodic command words (%QW\p.2.c\0.0 to %QW\p.2.c\0.0.25) and periodic control words (%IW\p.2.c\0.0 to %IW\p.2.c\0.0.25) are fully configurable by the user.

Their organisational structure is presented in three different ways, depending on how the gateway is used:

- Use of the standard configuration with 8 TeSys U motor starters;
- Configuration with PL7 PRO, using the standard profile FED C32 P;
- External configuration (with AbcConf), using the standard profile FED C32.

These objects are updated implicitly by the FIPIO master (by periodic FIPIO exchanges), at the start of the PLC cycle for the control words, and at the end of the PLC cycle for the command words.



ENGLISH

5.1.1. Standard Configuration with 8 TeSys U Motor Starters

The periodic command and control words described below correspond to the configuration described in Chapter 4 Software Implementation of the Gateway, page 25.

Inputs / Controls

FIPIO object	Description
%IW\p.2.c\0.0	Status of motor starter ①
%IW\p.2.c\0.0.1	Status of motor starter ②
%IW\p.2.c\0.0.2	Status of motor starter ③
%IW\p.2.c\0.0.3	Status of motor starter ④
%IW\p.2.c\0.0.4	Status of motor starter ⑤
%IW\p.2.c\0.0.5	Status of motor starter ⑥
%IW\p.2.c\0.0.6	Status of motor starter ⑦
%IW\p.2.c\0.0.7	Status of motor starter ⑧
%IW\p.2.c\0.0.8 …….. %IW\p.2.c\0.0.25	Not used (18 words)
%IW\p.2.c\0.0.26	Reserved (1 word)

Outputs / Commands

FIPIO object	Description
%QW\p.2.c\0.0	Command of motor starter ①
%QW\p.2.c\0.0.1	Command of motor starter ②
%QW\p.2.c\0.0.2	Command of motor starter ③
%QW\p.2.c\0.0.3	Command of motor starter ④
%QW\p.2.c\0.0.4	Command of motor starter ⑤
%QW\p.2.c\0.0.5	Command of motor starter ⑥
%QW\p.2.c\0.0.6	Command of motor starter ⑦
%QW\p.2.c\0.0.7	Command of motor starter ⑧
%QW\p.2.c\0.0.8 …… %QW\p.2.c\0.0.25	Not used (18 words)
%QW\p.2.c\0.0.26 %QW\p.2.c\0.0.27	Reserved (2 words)

5. FIPIO Objects Available for Programming

The input assigned to a motor starter gives the value of the status register for that motor starter (address 704 in the motor starter memory). The gateway handles the periodic Modbus communications with the motor starter in order to update the value of the FIPIO object that corresponds to the input.

The output value assigned to a motor starter is transmitted to that motor starter in order to update the value of its command register (address 455 in the motor starter memory). The gateway handles the periodic Modbus communications with the motor starter in order to transmit the value of the FIPIO object that corresponds to the output. Transmission is periodic, but writing a *new value* means that the gateway will transmit it as soon as possible.

N.B. If you delete any motor starters from the configuration (see Section 4.2.6 Deleting one or more TeSys U Motor Starters from the Configuration, page 31), you will need to adapt the two above tables accordingly. If, for example, you delete 3 motor starters (it has to be the last 3, whichever Modbus addresses are used), the statuses and commands for motor starters ⑥, ⑦ and ⑧ will become the words “Not used”.

5.1.2. Configuration under PL7 PRO with Profile FED C32 P

The actual contents of the periodic command and control words described below will depend entirely on the values you entered for the gateway’s configuration and adjustment parameters. These parameters are described in Chapter 9 Appendix B: LUFP1 Gateway Settings, page 100.

Periodic control words

FIPIO object	Description
%IW\p.2.c\0.0	Periodic control word n° 1
%IW\p.2.c\0.0.1	Periodic control word n° 2
……..	……
%IW\p.2.c\0.0.25	Periodic control word n°26
%IW\p.2.c\0.0.26	Reserved word

Periodic command words

FIPIO object	Description
%QW\p.2.c\0.0	Periodic command word n° 1
%QW\p.2.c\0.0.1	Periodic command word n° 2
……	……..
%QW\p.2.c\0.0.25	Periodic command word n°26
%QW\p.2.c\0.0.26	Reserved word
%QW\p.2.c\0.0.27	Reserved word

You can configure a total of 26 periodic words. If you configure 20 periodic control words, for example, you can then only configure a maximum of 6 periodic command words.

Periodic words (control and command) are organised in increasing order of the Modbus slaves (from n°1 to n°8). Within this schema, the words assigned to each slave are, in turn, organised in increasing order.

Example: In the example at the top of the next page, the gateway is configured to control and command 4 Modbus slaves. This example exploits the fact that the number of words is not necessarily the same for each slave.

5. FIPIO Objects Available for Programming

FIPIO object	Description (*example*)
%IW\p.2.c\0.0	Slave n°1: Control n°1
%IW\p.2.c\0.0.1	Slave n°1: Control n°2
%IW\p.2.c\0.0.2	Slave n°1: Control n°3
%IW\p.2.c\0.0.3	Slave n°2: Control n°1
%IW\p.2.c\0.0.4	Slave n°2: Control n°2
%IW\p.2.c\0.0.5	Slave n°3: Control n°1
%IW\p.2.c\0.0.6	Slave n°4: Control n°1
%IW\p.2.c\0.0.7	Slave n°4: Control n°2
%IW\p.2.c\0.0.8 …….. %IW\p.2.c\0.0.25	Not used (18 words)
%IW\p.2.c\0.0.26	Reserved

FIPIO object	Description (*example*)
%QW\p.2.c\0.0	Slave n°1: Command n°1
%QW\p.2.c\0.0.1	Slave n°2: Command n°1
%QW\p.2.c\0.0.2	Slave n°2: Command n°2
%QW\p.2.c\0.0.3	Slave n°2: Command n°3
%QW\p.2.c\0.0.4	Slave n°3: Command n°1
%QW\p.2.c\0.0.5	Slave n°3: Command n°2
%QW\p.2.c\0.0.6	Slave n°4: Command n°1
%QW\p.2.c\0.0.7 …… %QW\p.2.c\0.0.25	Not used (19 words)
%QW\p.2.c\0.0.26	Reserved
%QW\p.2.c\0.0.27	Reserved

5.1.3. Configuration under AbcConf with Profile FED C32

The actual contents of the periodic command and control words described below will depend entirely on the Modbus data / gateway memory associations you configured in AbcConf. These associations involve the “Data” or “Preset Data” elements in the Modbus query and response frames. See Chapter 7 Using ABC-LUFP Configurator, page 61, for the various possibilities offered by this gateway configuration tool, especially in conjunction with profile FED C32.

ENGLISH

Mapping of FIPIO PLC INPUTS to LUFP1 gateway memory

FIPIO object	Memory
%IW\p.2.c\0.0	16#0000-16#0001
%IW\p.2.c\0.0.1	16#0002-16#0003
%IW\p.2.c\0.0.2	16#0004-16#0005
%IW\p.2.c\0.0.3	16#0006-16#0007
%IW\p.2.c\0.0.4	16#0008-16#0009
%IW\p.2.c\0.0.5	16#000A-16#000B
%IW\p.2.c\0.0.6	16#000C-16#000D
%IW\p.2.c\0.0.7	16#000E-16#000F
%IW\p.2.c\0.0.8	16#0010-16#0011

FIPIO object	Memory
%IW\p.2.c\0.0.9	16#0012-16#0013
%IW\p.2.c\0.0.10	16#0014-16#0015
%IW\p.2.c\0.0.11	16#0016-16#0017
%IW\p.2.c\0.0.12	16#0018-16#0019
%IW\p.2.c\0.0.13	16#001A-16#001B
%IW\p.2.c\0.0.14	16#001C-16#001D
%IW\p.2.c\0.0.15	16#001E-16#001F
%IW\p.2.c\0.0.16	16#0020-16#0021
%IW\p.2.c\0.0.17	16#0022-16#0023

FIPIO object	Memory
%IW\p.2.c\0.0.18	16#0024-16#0025
%IW\p.2.c\0.0.19	16#0026-16#0027
%IW\p.2.c\0.0.20	16#0028-16#0029
%IW\p.2.c\0.0.21	16#002A-16#002B
%IW\p.2.c\0.0.22	16#002C-16#002D
%IW\p.2.c\0.0.23	16#002E-16#002F
%IW\p.2.c\0.0.24	16#0030-16#0031
%IW\p.2.c\0.0.25	16#0032-16#0033

Mapping of FIPIO PLC OUTPUTS to LUFP1 gateway memory

FIPIO object	Memory
%QW\p.2.c\0.0	16#0200-16#0201
%QW\p.2.c\0.0.1	16#0202-16#0203
%QW\p.2.c\0.0.2	16#0204-16#0205
%QW\p.2.c\0.0.3	16#0206-16#0207
%QW\p.2.c\0.0.4	16#0208-16#0209
%QW\p.2.c\0.0.5	16#020A-16#020B
%QW\p.2.c\0.0.6	16#020C-16#020D
%QW\p.2.c\0.0.7	16#020E-16#020F
%QW\p.2.c\0.0.8	16#0210-16#0211

FIPIO object	Memory
%QW\p.2.c\0.0.9	16#0212-16#0213
%QW\p.2.c\0.0.10	16#0214-16#0215
%QW\p.2.c\0.0.11	16#0216-16#0217
%QW\p.2.c\0.0.12	16#0218-16#0219
%QW\p.2.c\0.0.13	16#021A-16#021B
%QW\p.2.c\0.0.14	16#021C-16#021D
%QW\p.2.c\0.0.15	16#021E-16#021F
%QW\p.2.c\0.0.16	16#0220-16#0221
%QW\p.2.c\0.0.17	16#0222-16#0223

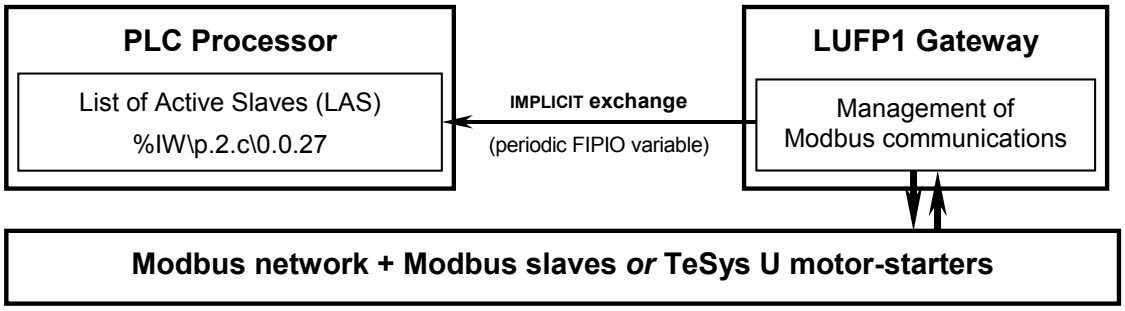
FIPIO object	Memory
%QW\p.2.c\0.0.18	16#0224-16#0225
%QW\p.2.c\0.0.19	16#0226-16#0227
%QW\p.2.c\0.0.20	16#0228-16#0229
%QW\p.2.c\0.0.21	16#022A-16#022B
%QW\p.2.c\0.0.22	16#022C-16#022D
%QW\p.2.c\0.0.23	16#022E-16#022F
%QW\p.2.c\0.0.24	16#0230-16#0231
%QW\p.2.c\0.0.25	16#0232-16#0233

Unlike with profile FED C32 P, you are not restricted to a maximum total of 26 periodic words. You can use all 26 input words and all 26 output words in the two above tables, organising them in the gateway memory as you see fit, within the limit of the addresses shown.

5. FIPIO Objects Available for Programming

5.2. List of Active Slaves (LAS) Service

This service consists of a single implicit input word (%IW\p.2.c\0.0.27) in which each bit (X0 to X7) corresponds to a Modbus slave. Each bit will have the value 1 if the corresponding Modbus slave is present and active on the Modbus network.



Bit	Standard configuration (8 TeSys Us)	Configuration under PL7 PRO (profile FED C32 P)	Configuration under AbcConf (profile FED C32)
X0	TeSys U motor starter n°1	Modbus slave n°1	1st Modbus slave
X1	TeSys U motor starter n°2	Modbus slave n°2	2nd Modbus slave
X2	TeSys U motor starter n°3	Modbus slave n°3	3rd Modbus slave
X3	TeSys U motor starter n°4	Modbus slave n°4	4th Modbus slave
X4	TeSys U motor starter n°5	Modbus slave n°5	5th Modbus slave
X5	TeSys U motor starter n°6	Modbus slave n°6	6th Modbus slave
X6	TeSys U motor starter n°7	Modbus slave n°7	7th Modbus slave
X7	TeSys U motor starter n°8	Modbus slave n°8	8th Modbus slave
X8-X15	Not used (bits at 0)		

In the event of no response, or an invalid response, from a Modbus slave (after the timeout and subsequent retransmissions of the Modbus command) the corresponding bit switches to 0. It will return to 1 as soon as the gateway receives a valid response from the slave.

If several periodic words (or several Modbus commands) are configured for the same Modbus slave, then it only requires one of the words to generate valid communications for the corresponding LAS bit to be set to 1.

You must configure at least one periodic command or control word per slave (profile FED C32 P) or at least one Modbus command per slave (AbcConf and profile FED C32). If not, the slave will be considered permanently absent (bit at 0).

If you want to communicate with a Modbus slave using the PKW service only, be aware that the LAS and PKW services are totally disassociated: a PKW response from a Modbus slave will not set to 1 the corresponding bit in word %IW\p.2.c\0.0.27.

Standard configuration (8 TeSys Us): If you delete any motor starters from the configuration (see Section 4.2.6 Deleting one or more TeSys U Motor Starters from the Configuration, page 31), you will need to adapt the above table accordingly. If, for example, you delete 3 motor starters (it has to be the last 3, whichever Modbus addresses are used), the bits corresponding to TeSys U motor starters n°6, 7 and 8—namely bits X5-X7—will take on the value “Not used (bits at 0)”.

ENGLISH

5. FIPIO Objects Available for Programming

Configuration under PL7 PRO (profile FED C32 P): The absence from the configuration of one or more Modbus slaves entails the same modifications as for the standard configuration of 8 TeSys Us.

Configuration under AbcConf (profile FED C32): The order of the slaves in the LAS word corresponds to the declaration order for the various “Nodes” in the “Sub-Network” element, as they appear in AbcConf. As in the two previous cases, if fewer than 8 Modbus slaves are used, then one or more slaves—starting with the 8th and working back towards the 1st—will be absent from the list.

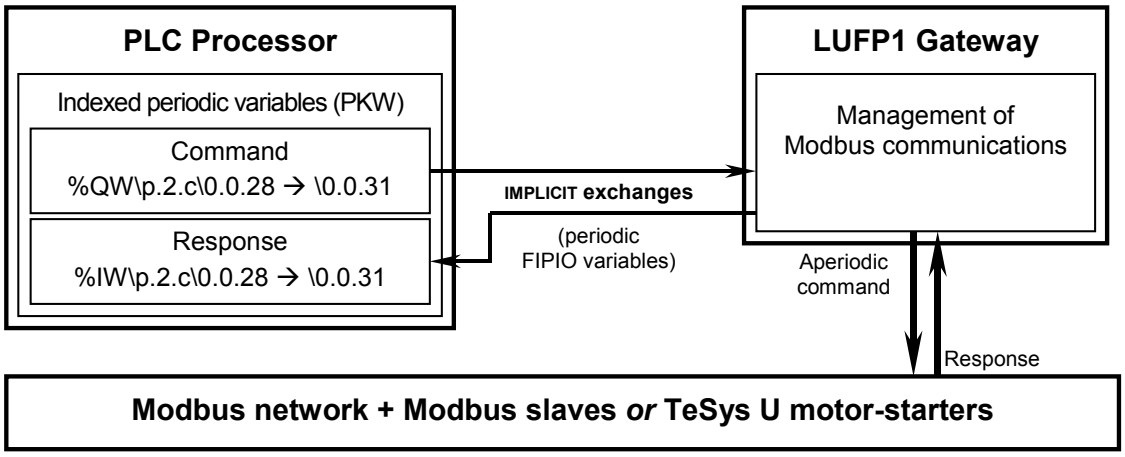
5.3. Indexed Periodic Variables (PKW) Service

5.3.1. Introduction to the PKW Service

With this service, you can configure and adjust any Modbus slave: it offers read and write access to any parameter of a slave via the periodic inputs and outputs of LUFP1 gateway. It can also be used for access to certain reserved registers on the LUFP1 gateway.

The PKW service is aperiodic on the Modbus network (a command is transmitted whenever one of the %QW outputs associated with the service changes) and periodic on the FIPIO network (implicit exchanges). You can use the service to perform the following tasks:

- Retrieving or updating the parameters on some or all of the Modbus slaves;
- Reading the value of one or more data for which a low refresh rate is sufficient;
- Reading the value of a data item when an event is signalled by a periodic variable.



In the absence of any response from the queried Modbus slave using this service, the gateway retransmits the same command 3 times in succession, waiting for 1 second between transmissions. If you wish, you can modify the values of these two parameters (number of retransmissions and timeout) using AbcConf (see Section 7.12.3 “Sub-Network” Element, page 93).

5.3.2. PKW Service: Command and Response

Indexed periodic variables (PKW) service: COMMAND

PLC output	Bits	Output name		Description
%QW\p.2.c\0.0.28	X0-X15	**PKE**	Parameter Kennung	Address of the datum to be read / written
%QW\p.2.c\0.0.29	X8-X15	**DN**	Device Number	Address of the queried slave: DN = 0 ..... 1st Modbus slave DN = 1 to 247 ..... *Address* of the queried Modbus slave DN = 254 ..... LUFP1 gateway DN = 255 ..... Broadcast to all Modbus slaves
	X0-X7	**R/W**	Read / Write	Read / write command code: R/W = 16#00 ..... Reset command and response R/W = 16#52 ..... Read a 16-bit word R/W = 16#53 ..... Read 2 consecutive 16-bit words *or* one 32-bit Dword R/W = 16#57 ..... Write a 16-bit word R/W = 16#58 ..... Write 2 consecutive 16-bit words *or* one 32-bit Dword
%QW\p.2.c\0.0.30	X0-X15	**PWE**	Parameter Wert	Value to be written (1st word): R/W = 16#57 ⇒ Value of the 16-bit word R/W = 16#58 ⇒ Value of the 1st 16-bit word *or* of the MSB word in the 32-bit word R/W = 16#•• ⇒ Value not used
%QW\p.2.c\0.0.31	X0-X15			Value to be written (2nd word): R/W = 16#58 ⇒ Value of the 2nd 16-bit word *or* of the LSB word in the 32-bit word R/W = 16#•• ⇒ Value not used

Indexed periodic variables (PKW) service: RESPONSE

PLC input	Bits	Input name		Description
%IW\p.2.c\0.0.28	X0-X15	**PKE**	Parameter Kennung	Address of the datum read / written
%IW\p.2.c\0.0.29	X8-X15	**DN**	Device Number	Address of the slave responding to the command: DN = 0 ..... 1st Modbus slave DN = 1 to 247 ..... *Address* of the Queried Modbus slave DN = 254 ..... LUFP1 gateway
	X0-X7	**R/W/N**	Read / Write / Error	Read / write command response code: R/W/N = 16#00 ..... Reset command and response R/W/N = 16#4E ..... Command runtime error R/W/N = 16#52 ..... Read a 16-bit word: OK R/W/N = 16#53 ..... Read 2 16-bit words: OK R/W/N = 16#57 ..... Write a 16-bit word: OK R/W/N = 16#58 ..... Write 2 16-bit words: OK
%IW\p.2.c\0.0.30	X0-X15	**PWE**	Parameter Wert	Value read (1st word): R/W/N = 16#4E ⇒ Specific error code (see description) R/W/N = 16#52 ⇒ Value of the 16-bit word R/W/N = 16#53 ⇒ Value of the 1st 16-bit word *or* of the MSB word in the 32-bit word R/W/N = 16#•• ⇒ Value not used (16#0000)
%IW\p.2.c\0.0.31	X0-X15			Value read (2nd word): R/W/N = 16#53 ⇒ Value of the 2nd 16-bit word *or* of the LSB word in the 32-bit word R/W/N = 16#•• ⇒ Value not used (16#0000)

ENGLISH

5.3.3. Using the Indexed Periodic Variables (PKW) Service

To perform a complete transaction, the FIPIO master must first update the whole of the command (%QW\p.2.c\0.0.28 to %QW\p.2.c\0.0.31), and then wait until the values in the response (%IW\p.2.c\0.0.28 to %IW\p.2.c\0.0.31) match the command (by comparing the PKE, DN and R/W—or, in the case of the response, R/W/N—values).

The gateway stores the response values for as long as the FIPIO master modifies *none* of the command values. If any output word in the PKW command (even the PWE outputs) is modified, a new command is transmitted over the Modbus network, except when R/W = 16#00.

- Stick strictly to the values given in the PKW service command table. Otherwise the gateway will send an erroneous response.
- Be careful not to write incorrect values in the outputs corresponding to the PKW service, as this would generate an incoherent command: only limited checks are run on the data used by this service, and it is therefore left to the FIPIO master PLC application to handle them.
- The PKE addresses corresponding to periodic command words (profile FED C32 P) or to registers assigned by Modbus write commands (profile FED C32) must not be used in PKW service write commands, as this could lead to a write conflict in the same registers.

  Such a command could however be envisaged if, for example, you wanted to run the same command on all the Modbus slaves *urgently and simultaneously* (DN = 255). In this case, you will also need to write the same value as the broadcast command PWE into the periodic command words that share the same PKE address, in order to avoid generating conflicts between the values transmitted.
- As the write commands (R/W = 16#57 and 16#58) return no value, you will have to use the appropriate read command (R/W = 16#52 or 16#53) if you want to check that the write operation has been performed correctly.

ENGLISH

Below are further details on the most important data in this service:

Device Number (DN):

DN = 0.....1st Modbus slave. This is TeSys U motor starter n°1, or Modbus slave n°1 when profile FED C32 P is used. When profile FED C32 is used, it stands for the first node configured in the “Sub-Network” element under AbcConf. In either case, the gateway will use the true physical Modbus address every time to query the “1st slave”.

DN = 1 to 247...Physical address of the queried Modbus slave. The queried slave *must* be one of the Modbus slaves configured under PL7 PRO (profile FED C32 P) or under AbcConf (profile FED C32). By querying one or more of the slaves episodically via the PKW service, you can avoid having to configure periodic words (under PL7 PRO) or Modbus commands (under AbcConf).

DN = 254.....LUFP1 gateway. With this reserved address you can read the values of certain registers on the gateway (see Section 5.3.5 LUFP1 Gateway Internal Registers, page 49). All the registers on the gateway are in 8-bit format and their values will be returned to bits 0-7 of word %IW\p.2.c\0.0.30. They cannot be written using the PKW service (prohibited operation). Only the 16-bit word read command (R/W = 16#52) can be used.

DN = 255.....Broadcast to all Modbus slaves. This DN value must only be used for write operations (R/W/N = 16#4E in the event of a read operation). On the Modbus network, the command generated uses the address 0: this means that all the Modbus slaves must accept the command, although none of them will acknowledge it.

The drawback with the broadcast command is that the gateway has no way of checking whether the Modbus slaves have received the command. Even if there are no slaves present on the Modbus network, the gateway will still return a valid PKW response (R/W/N = OK).

N.B. This feature can be particularly useful if you need to perform an urgent command on *all* the slaves *at the same time*.

DN = ●●●.....Incorrect address. Any value other than those indicated above will elicit an error code from the gateway (R/W/N = 16#4E).

Read / Write (R/W):

R/W = 16#00....Reset command and response. As long as R/W remains at 16#00, no command will be generated on the Modbus network (even if one of the other PKW service outputs is modified) and *all* the PKW service inputs will remain at 16#0000. Basically, you should use this value if you want to deactivate the PKW service.

N.B. As the PKW service only performs single read and write operations (the active command is only executed once), you could use this specific value of R/W to repeatedly overwrite the active R/W value. This would generate a series of identical commands (interspersed with resets to zero), thereby enabling the gateway to emulate the principle of permanent read and write operations.

R/W = 16#52....Read a 16-bit word. This is also the command to use for reading an 8-bit register from the gateway (DN = 254).

R/W = 16#53....Read 2 consecutive 16-bit words *or* a 32-bit Dword. This command is used mainly for reading the values of two 16-bit registers located at two consecutive addresses on the same Modbus slave. The value located at address PKE will be returned in the input %IW\p.2.c\0.0.30 (1st word of PWE) and the value at address PKE+1 in the input %IW\p.2.c\0.0.31 (2nd word of PWE).

N.B. This command should be preferred to the 16-bit word read command (R/W = 16#52) when using the PKW service to read a large number of consecutive registers on a Modbus slave.

R/W = 16#57....Write a 16-bit word. This command allows you to modify the value of the 16-bit register of which the address is given by the PKE output. Only the 1st word of the PWE output is used (%QW\p.2.c\0.0.30).

R/W = 16#58....Write 2 consecutive 16-bit words *or* a 32-bit Dword. This command is used mainly for modifying the value of two 16-bit registers located at two consecutive addresses on the same Modbus slave. The value of output %QW\p.2.c\0.0.30 (1st word of PWE) is written into address PKE and the value of output %QW\p.2.c\0.0.31 (2nd word of PWE) is written into address PKE+1.

N.B. This command should be preferred to the 16-bit word write command (R/W = 16#57) when using the PKW service to update a large number of consecutive registers on a Modbus slave.

Read / Write / Error (R/W/N):

R/W/N = 16#4E ⇒ This specific value enables the gateway to signal a PKW command runtime error to the FIPIO master. In this case, the gateway uses the LSB of the 1st PWE word in the response to return an error code to the FIPIO master:

- 16#04 if no Modbus response, or an erroneous Modbus response, is received.
- 16#01 to 16#08 (including 16#04) if an exception response is received from the Modbus slave. This value will correspond to the exception code (see Section 12.4 Modbus Protocol Exception Responses, page 122).

R/W/N = 16#●● ⇒ Copies the value of the R/W command register (see above), thereby also acknowledging the command.

5. FIPIO Objects Available for Programming

5.3.4. Sample Uses of PKW Service

The examples below illustrate commands addressed to Modbus slaves consisting mainly of the 8 TeSys U motor starters described in Chapter 4: Software Implementation of the Gateway (page 25).

- ***Sample reading of the value of a motor starter register (16-bit word)***

  The 1st fault register (address = 452) is read on TeSys U motor starter n°5 (address = 5).

  The result of the read operation is 16#0002 (presence of a magnetic fault only). Only the 1st PWE word in the response is used; the 2nd word is forced to 16#0000 by the gateway.

PKW Command

Output	Value	Meaning
%QW\p.2.c\0.0.28	16#01C4	PKE = 452
%QW\p.2.c\0.0.29	16#0552	DN = 16#05 R/W = 16#52 (read)
%QW\p.2.c\0.0.30	16#0000	PWE = 16#••••
%QW\p.2.c\0.0.31	16#0000	16#••••

PKW Response

Input	Value	Meaning
%IW\p.2.c\0.0.28	16#01C4	PKE = 452
%IW\p.2.c\0.0.29	16#0552	DN = 16#05 R/W/N = 16#52 (read OK)
%IW\p.2.c\0.0.30	16#0002	PWE = 16#0002 (used)
%IW\p.2.c\0.0.31	16#0000	16#0000 (not used)

- ***Sample writing of the value of a motor starter register (16-bit word)***

  The 2nd command register (address = 705) is written to TeSys U motor starter n°7 (address = 7) with the value 16#0006 (clear statistics + reset thermal memory).

  The result of the write operation is a simple acknowledgement, as no value is returned in the PWE of the response; instead of inserting a value, the gateway forces both words of the PWE to 16#0000.

PKW Command

Output	Value	Meaning
%QW\p.2.c\0.0.28	16#02C1	PKE = 705
%QW\p.2.c\0.0.29	16#0757	DN = 16#07 R/W = 16#57 (write)
%QW\p.2.c\0.0.30	16#0002	PWE = 16#0002
%QW\p.2.c\0.0.31	16#0000	16#••••

PKW Response

Input	Value	Meaning
%IW\p.2.c\0.0.28	16#02C1	PKE = 705
%IW\p.2.c\0.0.29	16#0757	DN = 16#07 R/W/N = 16#57 (write OK)
%IW\p.2.c\0.0.30	16#0000	PWE = 16#0000 (not used)
%IW\p.2.c\0.0.31	16#0000	16#0000 (not used)

- ***Sample reading of the value of two consecutive motor starter registers (16-bit words)***

  The operating duration (addresses = 119 for the MSB and 120 for the LSB) is read on TeSys U motor starter n°8 (address = 8).

  The result of the read operation (value of the register at 119 and value of the register at 120) is 16#000D and 16#C9C4 (total operating duration = 16#000DC9C4 = 903,620 seconds, or 251.0055 hours). Both of the response PWE words are therefore used.

PKW Command

Output	Value	Meaning
%QW\p.2.c\0.0.28	16#0077	PKE = 119
%QW\p.2.c\0.0.29	16#0853	DN = 16#08 R/W = 16#53 (read32)
%QW\p.2.c\0.0.30	16#0000	PWE = 16#••••
%QW\p.2.c\0.0.31	16#0000	16#••••

PKW Response

Input	Value	Meaning
%IW\p.2.c\0.0.28	16#0077	PKE = 119
%IW\p.2.c\0.0.29	16#0853	DN = 16#08 R/W/N = 16#53 (read32 OK)
%IW\p.2.c\0.0.30	16#000D	PWE = 16#000D (used)
%IW\p.2.c\0.0.31	16#C9C4	16#C9C4 (used)

ENGLISH

5. FIPIO Objects Available for Programming

- ***Sample writing of the value of two consecutive motor starter registers (16-bit words)***

  The timeout and threshold for the thermal reset (addresses = 607 and 608) are written to TeSys U motor starter n°2 (address = 2) with the following values: 90 s (16#005A) and 60% (16#003C).

  The result of the write operation is a simple acknowledgement, as no value is returned in the PWE of the response; instead of inserting one or two values, the gateway forces both words of the PWE to 16#0000.

PKW Command

Output	Value	Meaning
%QW\p.2.c\0.0.28	16#025F	PKE = 607
%QW\p.2.c\0.0.29	16#0258	DN = 16#02 R/W = 16#58 (write32)
%QW\p.2.c\0.0.30	16#005A	PWE = 16#005A
%QW\p.2.c\0.0.31	16#003C	16#003C

PKW Response

Input	Value	Meaning
%IW\p.2.c\0.0.28	16#025F	PKE = 607
%IW\p.2.c\0.0.29	16#0258	DN = 16#02 R/W/N = 16#58 (write32 OK)
%IW\p.2.c\0.0.30	16#0000	PWE = 16#0000 (not used)
%IW\p.2.c\0.0.31	16#0000	16#0000 (not used)

- ***Sample erroneous reading of the value of a motor starter register (16-bit word)***

  An attempt is made to read the status register value (address = 455 for a TeSys U motor starter) on a slave whose physical address corresponds to none of the addresses configured for the gateway's Modbus slaves (address = 32 = 16#20).

  The response is an error code (R/W/N = 16#4E) and is immediate: the gateway signals to the FIPIO master that the queried slave does not exist (1st word of PWE = error code = 16#0002).

PKW Command

Output	Value	Meaning
%QW\p.2.c\0.0.28	16#01C7	PKE = 455
%QW\p.2.c\0.0.29	16#2052	DN = 16#20 R/W = 16#52 (read)
%QW\p.2.c\0.0.30	16#0000	PWE = 16#••••
%QW\p.2.c\0.0.31	16#0000	16#••••

PKW Response

Input	Value	Meaning
%IW\p.2.c\0.0.28	16#01C7	PKE = 455
%IW\p.2.c\0.0.29	16#204E	DN = 16#20 R/W/N = 16#4E (ERROR)
%IW\p.2.c\0.0.30	16#0002	PWE = 16#0002 (error code)
%IW\p.2.c\0.0.31	16#0000	16#0000 (not used)

- ***Sample erroneous writing of the value of a motor starter register (16-bit word)***

  An attempt is made to write the status register value (address = 455) to TeSys U motor starter n°2 (address = 2) with the value 16#0001, but the register is in read-only mode.

  The response is an error code (R/W/N = 16#4E): the motor starter returns an exception response (1st word of PWE = error code = Modbus exception code = 16#0002 = Illegal Data Address).

PKW Command

Output	Value	Meaning
%QW\p.2.c\0.0.28	16#01C7	PKE = 455
%QW\p.2.c\0.0.29	16#0357	DN = 16#03 R/W = 16#57 (write)
%QW\p.2.c\0.0.30	16#0001	PWE = 16#0001
%QW\p.2.c\0.0.31	16#0000	16#••••

PKW Response

Input	Value	Meaning
%IW\p.2.c\0.0.28	16#01C7	PKE = 455
%IW\p.2.c\0.0.29	16#034E	DN = 16#03 R/W/N = 16#4E (ERROR)
%IW\p.2.c\0.0.30	16#0002	PWE = 16#0002 (exception code)
%IW\p.2.c\0.0.31	16#0000	16#0000 (not used)

ENGLISH

- ***Sample writing of the value of a register (16-bit word) on all the motor starters (broadcast)***

  The command register (address = 704) is written to all the TeSys U motor starters (address = broadcast = 255) with the value 16#2000 (Stop and Pause). **Caution:** this simultaneously switches off all the TeSys U motor starters, but you still need to insert the value 16#2000 in the periodic output words sent to the motor starters, otherwise they will return to their previous status at the next Modbus polling cycle.

  The result of the write operation is a simple acknowledgement, as no value is returned in the PWE of the response; instead of inserting a value, the gateway forces both words of the PWE to 16#0000.

PKW Command

Output	Value	Meaning
%QW\p.2.c\0.0.28	16#02C0	PKE = 704
%QW\p.2.c\0.0.29	16#FF57	DN = 16#FF R/W = 16#57 (write)
%QW\p.2.c\0.0.30	16#2000	PWE = 16#2000
%QW\p.2.c\0.0.31	16#0000	16#••••

PKW Response

Input	Value	Meaning
%IW\p.2.c\0.0.28	16#02C0	PKE = 704
%IW\p.2.c\0.0.29	16#FF57	DN = 16#FF R/W/N = 16#57 (write OK)
%IW\p.2.c\0.0.30	16#0000	PWE = 16#0000 (not used)
%IW\p.2.c\0.0.31	16#0000	16#0000 (not used)

- ***Sample reading of the value of a register (8-bit byte) on the LUFP1 gateway***

  The last error code from Modbus slave n°1 is read on the gateway (address = 300 and DN = 16#FE).

  If Modbus slave n°1 (TeSys U motor starter n°1 in the case of the standard configuration) has already been disconnected from the Modbus network, the gateway will have assigned it the error code 16#04. In this case, the result of the read operation is 16#••04. As the value of the register read is contained in the LSB of the 1st word in the response PWE, the corresponding MSB is forced to 16#00 by the gateway; likewise, the 2nd word is forced to 16#0000.

PKW Command

Output	Value	Meaning
%QW\p.2.c\0.0.28	16#012C	PKE = 300
%QW\p.2.c\0.0.29	16#FE52	DN = 16#FE R/W = 16#52 (read)
%QW\p.2.c\0.0.30	16#0000	PWE = 16#••••
%QW\p.2.c\0.0.31	16#0000	16#••••

PKW Response

Input	Value	Meaning
%IW\p.2.c\0.0.28	16#012C	PKE = 300
%IW\p.2.c\0.0.29	16#FE52	DN = 16#FE R/W/N = 16#52 (read OK)
%IW\p.2.c\0.0.30	16#0004	PWE = 16#0004 (LSB used)
%IW\p.2.c\0.0.31	16#0000	16#0000 (not used)

ENGLISH

5.3.5. LUFP1 Gateway Internal Registers

The registers presented in this section are internal to the gateway and are all in 8-bit format. The PKW service provides access to them (DN = 254), but only with the aid of a 16-bit word read command. The gateway inserts the result of a read operation in bits 0-7 of the 1st word in the PWE (%QW\p.2.c\0.0.30); the rest of the PWE is forced to 0.

The “RO” access right means that the registers are in “read-only” mode, rather than “read / write” (“R/W”). But even when a register is in “R/W”, the gateway can reject a write operation via the PKW service if the value of the PWE in the write command is incorrect. This tends to happen with any value other than 16#0000 in the case of registers 300 to 307 (error code reset).

5. FIPIO Objects Available for Programming

LUFP1 Gateway Internal Registers: Summary

Addresses	Rights	Registers
Adjustment parameters (%MW\p.2.c\0.0.20 to %MW\p.2.c\0.0.49)		
0 to 9	RO	General Modbus configuration for the gateway
10 to 21	R/W	Cycle time settings for command and control words
22 to 29	RO	Physical addresses of Modbus slaves
30 to 59	RO	Registers not used (16#••)
Configuration parameters (%KW\p.2.c\0.0 to %KW\p.2.c\0.0.29)		
60 to 67	RO	Number of periodic command and control words
68 to 119	R/W	Addresses of each of the command and control words
Registers not used		
120 to 299	RO	Not used (values equal to 16#00)
Error codes relating to Modbus slaves n°1 to 8		
300 to 307	R/W	Error codes relating to Modbus slaves n°1 to 8 (last error)

The adjustment and configuration parameters will only be present when the gateway is configured using PL7 PRO, i.e. in accordance with the standard profile FED C32 P. You should therefore not use addresses 0 to 119 if you configure the gateway with AbcConf (profile FED C32 cannot be configured under PL7 PRO).

ENGLISH

These parameters are shown below with the sole purpose of listing their addresses and access rights to the PKW service. For more details about them, see Appendix B: LUFP1 Gateway Settings (Chapter 9, page 100). To make it easier to read, the table below is based on the FIPIO objects that correspond to the gateway's configuration and adjustment parameters:

LUFP1 Gateway Internal Registers: Details

FIPIO object	Bits 0 to 7 (LSB)			Bits 8 to 15 (MSB)		
	Add.	Rights	Name	Add.	Rights	Name
%MW\p.2.c\0.0.20	0	RO	Modbus speed	1	RO	Modbus format
%MW\p.2.c\0.0.21	2	RO	Modbus response timeout	3	RO	Number of retransmissions
%MW\p.2.c\0.0.22	4	RO	Modbus reconnection time	5	RO	Not used
%MW\p.2.c\0.0.23	6	RO	Behaviour on loss of FIPIO	7	RO	Not used
%MW\p.2.c\0.0.24	8	RO	Not used	9	RO	Not used
%MW\p.2.c\0.0.25	10	R/W	Fast cycle: Commands	11	R/W	Normal cycle: Commands
%MW\p.2.c\0.0.26	12	R/W	Fast cycle: Controls	13	R/W	Normal cycle: Controls
%MW\p.2.c\0.0.27	14	R/W	Speed: Commands 1 to 8	15	R/W	Speed: Commands 9 to 16
%MW\p.2.c\0.0.28	16	R/W	Speed: Commands 17 to 24	17	R/W	Speed: Commands 25 to 26
%MW\p.2.c\0.0.29	18	R/W	Speed: Controls 1 to 8	19	R/W	Speed: Controls 9 to 16
%MW\p.2.c\0.0.30	20	R/W	Speed: Controls 17 to 24	21	R/W	Speed: Controls 25 to 26
%MW\p.2.c\0.0.31	22	RO	Address of Modbus slave n°1	23	RO	Address of Modbus slave n°2
%MW\p.2.c\0.0.32	24	RO	Address of Modbus slave n°3	25	RO	Address of Modbus slave n°4
%MW\p.2.c\0.0.33	26	RO	Address of Modbus slave n°5	27	RO	Address of Modbus slave n°6
%MW\p.2.c\0.0.34	28	RO	Address of Modbus slave n°7	29	RO	Address of Modbus slave n°8
%MW\p.2.c\0.0.35	30	RO	Not used	31	RO	Not used
•••••••••••••	••	•••	••••••••	••	•••	••••••••
%MW\p.2.c\0.0.49	58	RO	Not used	59	RO	Not used

5. FIPIO Objects Available for Programming

FIPIO object	Bits 0 to 7 (LSB)			Bits 8 to 15 (MSB)		
	Add.	Rights	Name	Add.	Rights	Name
%KW\p.2.c\0.0	60	RO	Number of commands: slaves 1-2	61	RO	Number of commands: slaves 3-4
%KW\p.2.c\0.0.1	62	RO	Number of commands: slaves 5-6	63	RO	Number of commands: slaves 7-8
%KW\p.2.c\0.0.2	64	RO	Number of controls: slaves 1-2	65	RO	Number of controls: slaves 3-4
%KW\p.2.c\0.0.3	66	RO	Number of controls: slaves 5-6	67	RO	Number of controls: slaves 7-8
	Bits 8 to 15 (MSB)			Bits 0 to 7 (LSB)		
%KW\p.2.c\0.0.4	68	R/W	Com/con address n° 1 (MSB)	69	R/W	Com/con address n° 1 (LSB)
%KW\p.2.c\0.0.5	70	R/W	Com/con address n° 2 (MSB)	71	R/W	Com/con address n° 2 (LSB)
%KW\p.2.c\0.0.6	72	R/W	Com/con address n° 3 (MSB)	73	R/W	Com/con address n° 3 (LSB)
%KW\p.2.c\0.0.7	74	R/W	Com/con address n° 4 (MSB)	75	R/W	Com/con address n° 4 (LSB)
%KW\p.2.c\0.0.8	76	R/W	Com/con address n° 5 (MSB)	77	R/W	Com/con address n° 5 (LSB)
%KW\p.2.c\0.0.9	78	R/W	Com/con address n° 6 (MSB)	79	R/W	Com/con address n° 6 (LSB)
%KW\p.2.c\0.0.10	80	R/W	Com/con address n° 7 (MSB)	81	R/W	Com/con address n° 7 (LSB)
%KW\p.2.c\0.0.11	82	R/W	Com/con address n° 8 (MSB)	83	R/W	Com/con address n° 8 (LSB)
%KW\p.2.c\0.0.12	84	R/W	Com/con address n° 9 (MSB)	85	R/W	Com/con address n° 9 (LSB)
%KW\p.2.c\0.0.13	86	R/W	Com/con address n°10 (MSB)	87	R/W	Com/con address n°10 (LSB)
%KW\p.2.c\0.0.14	88	R/W	Com/con address n°11 (MSB)	89	R/W	Com/con address n°11 (LSB)
%KW\p.2.c\0.0.15	90	R/W	Com/con address n°12 (MSB)	91	R/W	Com/con address n°12 (LSB)
%KW\p.2.c\0.0.16	92	R/W	Com/con address n°13 (MSB)	93	R/W	Com/con address n°13 (LSB)
%KW\p.2.c\0.0.17	94	R/W	Com/con address n°14 (MSB)	95	R/W	Com/con address n°14 (LSB)
%KW\p.2.c\0.0.18	96	R/W	Com/con address n°15 (MSB)	97	R/W	Com/con address n°15 (LSB)
%KW\p.2.c\0.0.19	98	R/W	Com/con address n°16 (MSB)	99	R/W	Com/con address n°16 (LSB)
%KW\p.2.c\0.0.20	100	R/W	Com/con address n°17 (MSB)	101	R/W	Com/con address n°17 (LSB)
%KW\p.2.c\0.0.21	102	R/W	Com/con address n°18 (MSB)	103	R/W	Com/con address n°18 (LSB)
%KW\p.2.c\0.0.22	104	R/W	Com/con address n°19 (MSB)	105	R/W	Com/con address n°19 (LSB)
%KW\p.2.c\0.0.23	106	R/W	Com/con address n°20 (MSB)	107	R/W	Com/con address n°20 (LSB)
%KW\p.2.c\0.0.24	108	R/W	Com/con address n°21 (MSB)	109	R/W	Com/con address n°21 (LSB)
%KW\p.2.c\0.0.25	110	R/W	Com/con address n°22 (MSB)	111	R/W	Com/con address n°22 (LSB)
%KW\p.2.c\0.0.26	112	R/W	Com/con address n°23 (MSB)	113	R/W	Com/con address n°23 (LSB)
%KW\p.2.c\0.0.27	114	R/W	Com/con address n°24 (MSB)	115	R/W	Com/con address n°24 (LSB)
%KW\p.2.c\0.0.28	116	R/W	Com/con address n°25 (MSB)	117	R/W	Com/con address n°25 (LSB)
%KW\p.2.c\0.0.29	118	R/W	Com/con address n°26 (MSB)	119	R/W	Com/con address n°26 (LSB)
	120	RO	Not used (16#00)	121	RO	Not used (16#00)
	• • •	• • •	• • • • • • • • • • • • • • •	• • •	• • •	• • • • • • • • • • • • • • •
	298	RO	Not used (16#00)	299	RO	Not used (16#00)
	300	R/W	Error code: slave n°1	301	R/W	Error code: slave n°2
	302	R/W	Error code: slave n°3	303	R/W	Error code: slave n°4
	304	R/W	Error code: slave n°5	305	R/W	Error code: slave n°6
	306	R/W	Error code: slave n°7	307	R/W	Error code: slave n°8

N.B. By modifying the values of the registers at addresses 68 to 119, the FIPIO master can alter the very nature of the information exchanged between the gateway and the slaves. **Such operations therefore require the utmost care!**

Slave error codes: Each of the registers at addresses 300 to 307 contains the code of the last error declared by the gateway for the Modbus slave concerned. If the gateway receives an exception response from a Modbus slave (see Section 12.4 Modbus Protocol Exception Responses, page 122) during periodic exchanges, it puts the exception code in the corresponding register (300-307). The value of each of these registers remains the same until either a new error is signalled for the slave concerned, or you reset it to zero using the PKW service (the gateway only accepts write commands where PWE = 16#0000).

The value of the “error code” is 16#04 if the corresponding slave is declared absent by the gateway (after the Modbus periodic commands for the slave have been retransmitted with timeouts).

ENGLISH

5. FIPIO Objects Available for Programming

N.B. The LAS service is updated at the same time as registers 300 to 307 when this type of event occurs (see Section 5.2 List of Active Slaves (LAS) Service, page 42). You can therefore use the various bits in the periodic word of this service (%IW\p.2.c\0.0.27) to trigger a read operation on the internal register assigned to the slave declared absent by the LAS service.

5.4. FIPIO Diagnostic Objects

We strongly recommend that you adhere to the following five rules while using the other FIPIO objects associated with the LUFP1 gateway:

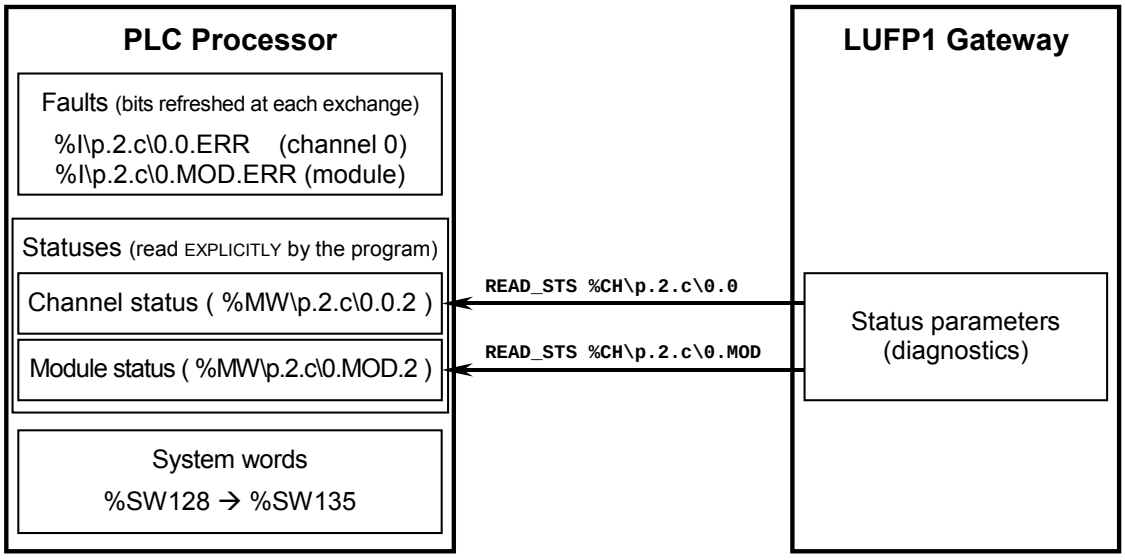
① Check system words %SW128 to %SW135
Each bit in this group of words signals the status of a device connected to the FIPIO bus (addresses 0 to 127). See Section 5.4.1 System Words %SW128 to %SW135, page 53, for a detailed description of these system words.

② Check the implicit "channel fault" bit (%I\p.2.c\0.0.ERR) that monitors the connection point
This bit, normally at 0, switches to 1 in the event of a fault or break in the FIPIO connection. If so, none of the %IW\p.2.c\0.0.•• inputs will be valid.

③ If there is a “channel fault” (see point ②), do an explicit read on the “Channel status” (%MW\p.2.c\0.0.2)
This information is updated by the explicit command `READ_STS %CH\p.2.c\0.0;` . See Section 5.4.2 Channel Status (%MW\p.2.c\0.0.2), page 53, for a description of the “Channel status” bits.

④ Check the implicit "module fault" bit (%I\p.2.c\0.MOD.ERR) that monitors the gateway
As with the “Channel fault”, this bit is normally at 0 but switches to 1 in the event of a fault or break in the FIPIO connection.

⑤ If there is a “module fault” (see point ④), do an explicit read on the “Module status” (%MW\p.2.c\0.MOD.2)
This information is updated by the explicit command `READ_STS %CH\p.2.c\0.MOD;` . See Section 5.4.3 Module Status (%MW\p.2.c\0.MOD.2), page 53, for a description of the “Module status” bits.



See also the PL7 PRO documentation or on-line help (Working with communications > FIPIO bus communications > Programming a FIPIO communication > Managing FIPIO faults) to learn more about the procedures for managing FIPIO faults.

5. FIPIO Objects Available for Programming

5.4.1. System Words %SW128 to %SW135

Each bit in this group of words signals the status of a device connected to the FIPIO bus (addresses 0 to 127). These bits are normally at 1, but switch to 0 in the event of a fault on the connection point (faulty FIPIO connection point). For a non-configured connection point, the corresponding bit is always at 1.

The table below maps the correspondences between these system word bits and the addresses on the FIPIO bus:

	X0	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15
%SW128:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
%SW129:	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
%SW130:	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
%SW131:	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
%SW132:	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
%SW133:	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
%SW134:	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
%SW135:	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127

5.4.2. Channel Status (%MW\p.2.c\0.0.2)

Each of the “channel status” bits corresponds to a fault generated either by the gateway or by the PLC. A fault is present if the corresponding bit is at 1. The “channel status” is updated by the explicit command `READ_STS %CH\p.2.c\0.0;` .

	Bits	Meaning
Channel status managed by the gateway	X0-X3	Reserved (bits at 0)
	X4	**Internal fault:** The gateway is considered absent. It may be out of supply or absent from the FIPIO network. If it was configured using the standard profile FED C32, it may be that an unauthorised memory address was used under AbcConf (see the warnings on pages 72 and 76).
	X5	**Hardware configuration fault:** The standard FIPIO profile selected for configuring the gateway under PL7 PRO was not the right one. This fault is signalled when profile FED C32 is used in place of profile FED C32 P, and vice versa.
	X6	**Communication fault with the PLC**
	X7	**Application fault:** The configuration and adjustment parameters defined for profile FED C32 P, which corresponds to the gateway, contain incorrect values.
Channel status managed by the PLC	X8-X15	Reserved (bits at 0)

5.4.3. Module Status (%MW\p.2.c\0.MOD.2)

Each of the “module status” bits corresponds to a specific fault. A fault is present if the corresponding bit is at 1. The “module status” is updated by the explicit command `READ_STS %CH\p.2.c\0.MOD;` .

Bit	Meaning
X0	Reserved (bit at 0)
X1	**Functional fault:** Fault signalled by the gateway; reading “channel status” recommended (see above).
X2-X4	Reserved (bit at 0)

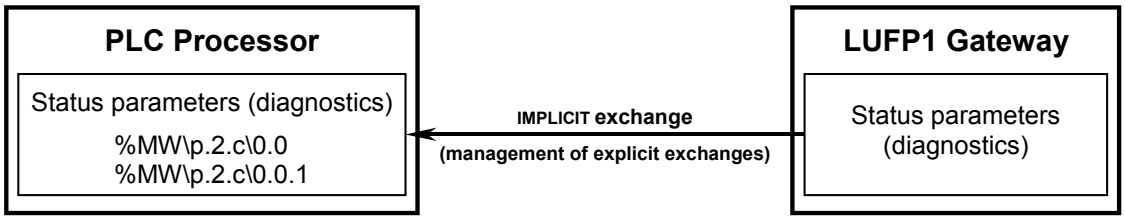
Bit	Meaning
X5	**Hardware configuration fault:** The standard FIPIO profile configured under PL7 PRO was the wrong one.
X6	**Gateway absent:** Check for an “internal fault” in the “channel status” (see above).
X7-X15	Reserved (bits at 0)

ENGLISH

5. FIPIO Objects Available for Programming

5.5. Status of Explicit Exchanges

The two implicit words %MW\p.2.c\0.0 (exchange in progress) and %MW\p.2.c\0.0.1 (report) serve to control explicit exchanges. You should use them if you need to program explicit exchanges in the FIPIO master PLC application.



Exchange in progress (%MW\p.2.c\0.0)

Bit	Meaning
X0	Reading status (if bit at 1) Make sure this bit is at 0 before you perform a new explicit exchange!
X1	Reserved (bit at 0)
X2	Sending adjustment parameters (if bit at 1)
X3-X14	Reserved (bits at 0)
X15	Sending configuration parameters (if bit at 1)

Report (%MW\p.2.c\0.0.1)

Bit	Meaning
X0	Read status: OK (if bit at 0)
X1	Reserved (bit at 0)
X2	Adjustment parameters received and accepted by gateway (if bit at 0)
X3-X14	Reserved (bits at 0)
X15	Configuration parameters received and accepted by gateway (if bit at 0)

Before carrying out an exchange, check that the relevant bit (%MW\p.2.c\0.0:X••) is at 0 (exchange inactive). The bit will switch to 1 for the duration of the exchange.

When the exchange is complete (the bit switches back to 0), you can check the bit at the other end (%MW\p.2.c\0.0.1:X••) to determine whether the exchange succeeded (bit at 0) or failed (bit at 1).

5.6. Parameters Specific to Profile FED C32 P

The gateway configuration and adjustment parameters are only stored in the FIPIO master PLC memory if the gateway was configured (under PL7 PRO) to support configuration and adjustment using standard profile FED C32 P.

If the gateway is configured with standard profile FED C32, then these parameters do not exist! Configuration and adjustment must then be carried out independently. In this case, the configuration tool “ABC-LUFP Configurator” will be required: see Chapter 7 Using ABC-LUFP Configurator, page 61. The software implementation of the gateway under PL7 PRO, using profile FED C32, is described in Chapter 6 Advanced Implementation of the Gateway, page 56. From the viewpoint of PL7 PRO, it then becomes a “locally configured” gateway.

Some of the configuration and adjustment parameters cannot be modified after the initial setup of the gateway using the PLC.
See the tables in Chapters 9.1 Configuration Parameters and 9.2 Adjustment Parameters.

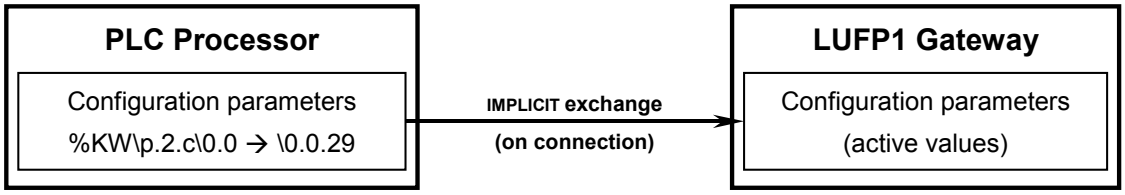
ENGLISH

5. FIPIO Objects Available for Programming

5.6.1. Configuration Parameters (%KW\p.2.c\0.0 to %KW\p.2.c\0.0.29)

The 30 configuration parameters of profile FED C32 P are transmitted to the gateway implicitly when it connects up to the FIPIO network.

When you modify them in ONLINE mode under PL7 PRO and save the changes, communications in channel 0 are paused while the new settings are sent to the gateway.



These parameters are described in Appendix B: LUFP1 Gateway Settings (Chapter 9, page 100).

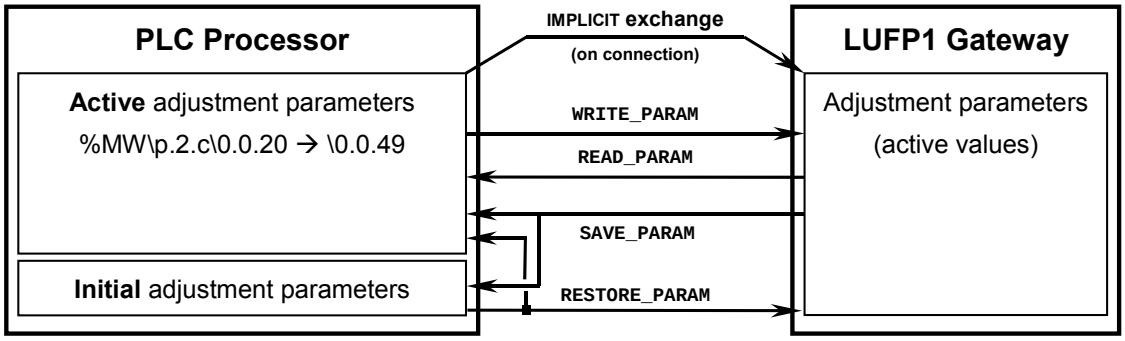
5.6.2. Adjustment Parameters (%MW\p.2.c\0.0.20 to %MW\p.2.c\0.0.49)

The 30 adjustment parameters of profile FED C32 P are transmitted to the gateway implicitly when it connects up to the FIPIO network.

When you modify them in ONLINE mode under PL7 PRO and save the changes, a command is generated to send the parameters to the gateway.

You can also use the following functions to perform explicit exchanges between the FIPIO PLC and the LUFP1 gateway:

- READ_PARAM %CH\p.2.c\0.0..... Read the adjustment parameters on the gateway.
- WRITE_PARAM %CH\p.2.c\0.0..... Write adjustment parameters to the gateway.
- SAVE_PARAM %CH\p.2.c\0.0..... Save the gateway adjustment parameters; the values replace the initial adjustment parameters.
- RESTORE_PARAM %CH\p.2.c\0.0 .... Restore the initial adjustment parameters.



N.B. The values of the initial adjustment parameters are those defined using the PL7 PRO configuration editor or, alternatively, the most recently saved values.

These parameters are described in Appendix B: LUFP1 Gateway Settings (Chapter 9, page 100).

ENGLISH

6. Advanced Implementation of the Gateway

This chapter marks the start of **Part II** of the LUFP1 Gateway User Manual. This part comprises Chapters 6 and 7 and focuses on the use of "ABC-LUFP Configurator". This configuration tool enables the gateway to be configured and adjusted independently of PL7 PRO. The possibilities offered by the tool go well beyond the limits imposed by the standard profile FED C32 P. The whole of Part II therefore supplements Chapter 4 Software Implementation of the Gateway, page 25, and supersedes certain sections.

6.1. Introduction to Advanced Implementation of the Gateway

The Advanced Implementation of the Gateway gets around certain limits imposed by the Software Implementation of the Gateway (Chapter 4, page 25). The main differences between the two implementation methods are summarised in the table below:

	Software Implementation of the Gateway (Chapter 4)	**Advanced Implementation of the Gateway (Chapter 6)**
FIPIO Profile	Standard profile "**FED C32 P**"	Standard profile "**FED C32**"
Modbus slaves	Maximum of 8 Modbus slaves	
Configuration method	Configuration and adjustment under PL7 PRO These parameters are downloaded to the gateway by the PLC on FIPIO connection and / or at the request of the FIPIO master (by explicit exchange)	Configuration performed entirely in ABC-LUFP Configurator The download is performed by ABC-LUFP Configurator and the configuration is stored in the gateway's static memory
Periodicity of Modbus exchanges	1 fast cycle and 1 normal cycle for write commands Ditto for read commands	Each Modbus command has its own cycle time (see below)
Modbus command transfer modes	Commands: Periodic mode, with immediate transmission following any change in the command value Controls: Periodic mode	Periodic or aperiodic communications: each Modbus command is configured individually
Number of Modbus commands	Maximum of 26 Modbus commands (whether read or write commands)	
Number of words read / written per Modbus command	Only 1 word can be read / written for each command configured	The number of words read / written by the same Modbus command is defined at the config stage, but the maximum (N max) depends on the Modbus slave
Configuration of Modbus command degraded modes	All Modbus commands are configured in the same way	Each Modbus command has its own configuration
FIPIO setup objects	Adjustment..... %MW\p.2.c\0.0.20 to %MW\p.2.c\0.0.49 Configuration ... %KW\p.2.c\0.0 to %KW\p.2.c\0.0.29	These FIPIO objects do not exist, and the corresponding internal registers are therefore not accessible via the gateway's PKW service

The main advantage of ABC-LUFP Configurator (AbcConf) lies in the possibility of configuring the gateway's "Modbus scanner" more precisely than under PL7 PRO, in that each Modbus command has its own configuration.

However, using this implementation method for the LUFP1 gateway substantially modifies the operating principle described and illustrated in Chapter 7 Using ABC-LUFP Configurator, page 61. The differences are set out below:

ENGLISH

6. Advanced Implementation of the Gateway

⇨ The exchanges between the gateway (acting as Modbus master) and the Modbus slaves are configured entirely using “ABC-LUFP Configurator”. This configuration tool really goes into the details (exchange timeouts, communication modes, frame content, etc.), which makes it somewhat harder to use. A chapter of this guide has therefore been given over to the subject (Chapter 7 Using ABC-LUFP Configurator, page 61).

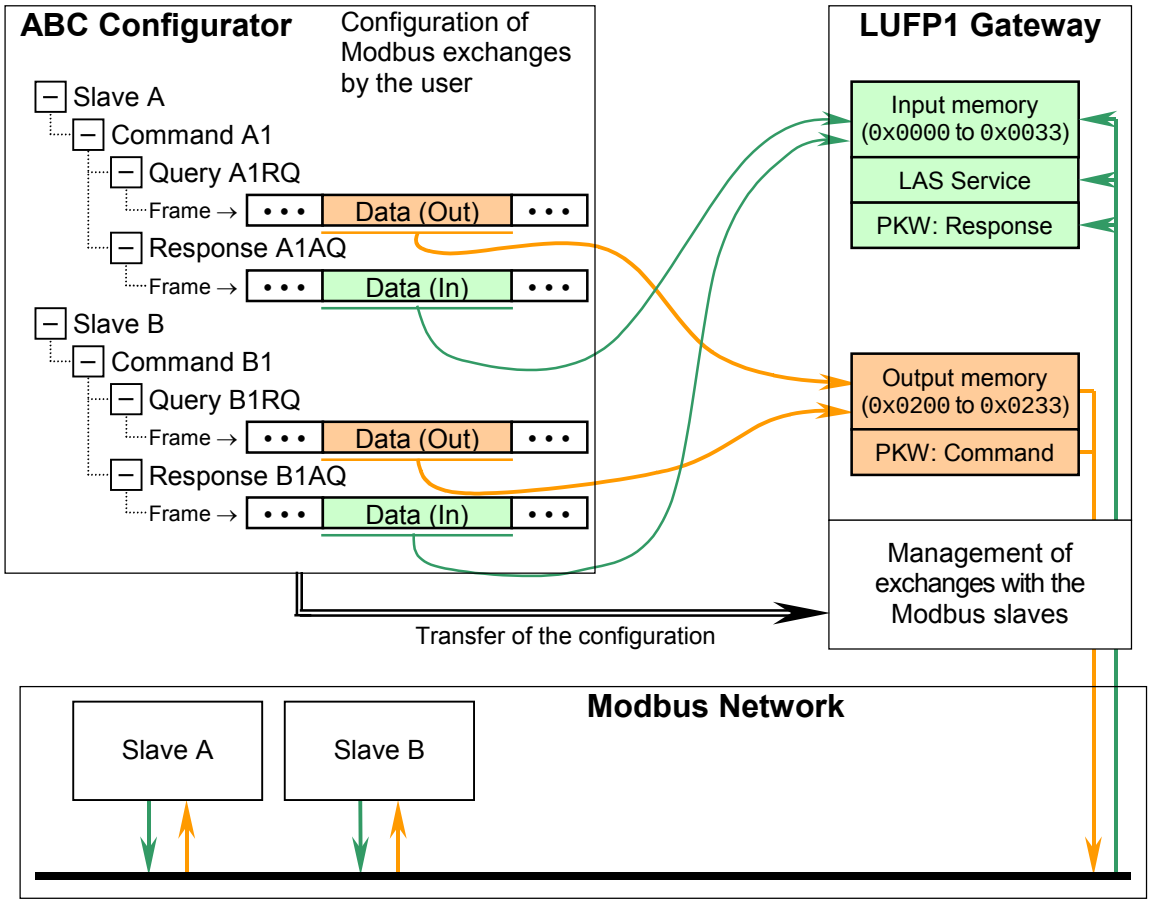
By configuring Modbus command queries and responses with this tool, the user creates links between part of the contents of the Modbus frames and the contents of the gateway’s physical memory (input memory for the contents of the Modbus responses and output memory for the contents of the queries). **N.B.** Only use the address ranges indicated in this guide when creating these links.

⇨ The periodic exchanges between the FIPIO master PLC and the LUFP1 gateway are limited to transferring the contents of the gateway’s input memory to the %IW inputs on the PLC and transferring the PLC’s %QW outputs to the gateway’s output memory.

The FIPIO network is totally separate from the Modbus network. The frames on a network are not directly “translated” by the gateway to generate frames on the other network. Instead, the exchanges between the contents of the gateway’s memory and the Modbus slaves make up a system independent of the one entrusted with managing the exchanges between this gateway memory and the FIPIO master.

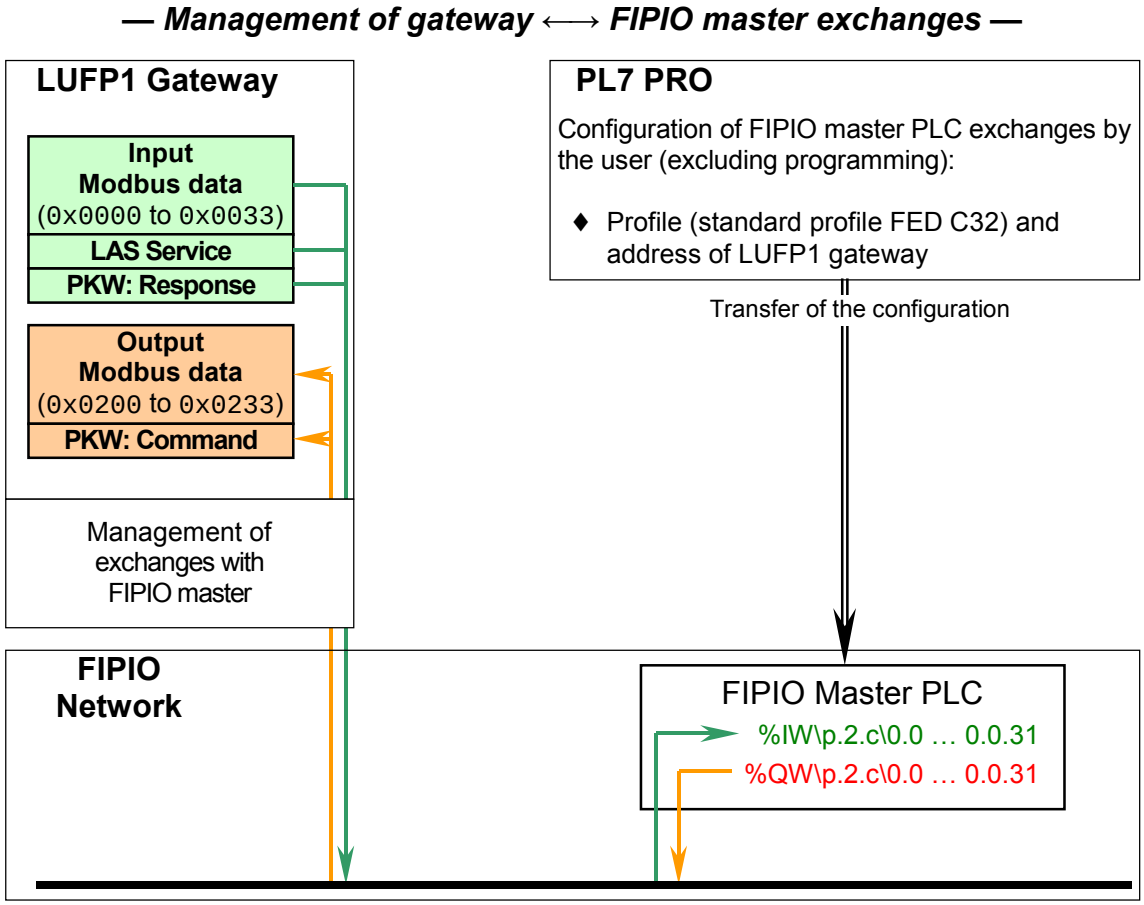
The two synopses that follow illustrate the independent management of each of the two networks:

— Management of gateway ⟷ Modbus slave exchanges —



ENGLISH

6. Advanced Implementation of the Gateway



6.2. Implementation under PL7 PRO

The implementation described below is based on the one in Section 4.2 Gateway Configuration under PL7 PRO, page 27. We just present a summary of that implementation here, along with the modifications entailed by using profile FED C32 instead of FED C32 P:

A configuration example can be found on the CD: LU9CD1 "`LUFP1_FEDC32_Example.cfg`". The system architecture for this configuration is identical in all respects to the one described in Chapter 4 Software Implementation of the Gateway (page 25). The example is provided mainly to assist you in understanding the uses and possibilities of AbcConf.

- **Adding the LUFP1 Gateway under PL7 PRO**.....Section 4.2.2 ... Page 27
  ⇒ Select basic module "FED C32" instead of basic module "FED C32 P".
- **Configuring and Adjusting the LUFP1 Gateway**.....Section 4.2.3 ... Page 28
  ⇒ Do not double-click on basic module `0 | FED C32`, as the gateway cannot be configured under PL7 PRO.
  ⇒ Ignore the description of how configuration and adjustment parameters are downloaded from the FIPIO master PLC to the LUFP1 gateway. When AbcConf is used—and the configuration thus downloaded from AbcConf (see next steps)—the parameters will be located on the gateway only. From the viewpoint of PL7 PRO and the FIPIO master PLC, the gateway will have a "local configuration".
- **Configuration Parameter Values, Standard Config**.....Section 4.2.4 ... Page 29
  ⇒ Ignore this step for the time being; it describes elements which, under AbcConf, will come into play later on.

ENGLISH

6. Advanced Implementation of the Gateway

- **Adjustment Parameter Values, Standard Configuration**.....Section 4.2.5 ... Page 30
  ⇒ Ignore this step for the time being; it describes elements which, under AbcConf, will come into play later on.

- **Deleting one or more TeSys U Motor Starters from the Configuration**.....Section 4.2.6 ... Page 31
  ⇒ This operation—described in Section 7.7 Deleting a Modbus Slave, page 66—is specific to AbcConf. You must therefore run AbcConf and load the example file provided in order to delete a motor starter.

- **Configuring Gateway Inputs / Outputs** .....Section 4.2.7 ... Page 32
  ⇒ As with PL7 PRO, no operation need be performed if you use the example file provided (this section describes the inputs / outputs corresponding to the configuration entered during the previous steps). If you want to modify the configuration or create a new one, consult the relevant sections of Chapter 7 Using ABC-LUFP Configurator, page 61. The explanations that follow seek to describe the configuration illustrated in the example.

  ⇒ The gateway inputs / outputs are configured entirely under AbcConf. The Modbus frames configured for the various slaves (or "Nodes") of the Modbus network (or "Sub-Network") include "Data" or "Preset Data" elements. Each element is a link between the contents of the Modbus frame it represents and a precise location in the gateway memory. You can configure the size of the data exchanged, and the memory location for the exchanges. This allows you to organise the gateway inputs and outputs as you see fit, within the limit of the addresses that correspond to the inputs / outputs (see Section 5.1.3 Configuration under AbcConf with Profile FED C32, page 41).

  ⇒ The configuration example provided with the gateway, "`LUFP1_FEDC32_Example.cfg`", is designed to reproduce the standard configuration used in the gateway implementation with profile FED C32 P (8 TeSys U motor starters). The configuration of the gateway inputs / outputs is therefore identical to the one described for the standard configuration (the two tables on page 33). Each "TeSys U n°•" node in the example comprises a read command from the TeSys U status register (address = 455 = 16#01C7) and a write command from the TeSys U command register (address = 704 = 0x2C0). These commands are configured in exactly the same way, except for their "Data" elements. The "Modbus frame content / gateway memory address" links for these elements, on which the equivalence with the standard configuration is based, are configured as follows:

Complete tree structure of the AbcConf element containing the link	Data length	Data location
Sub-Network → TeSys U n°1 → Preset Multiple Registers → Query → Data	0x0002	0x0200
Sub-Network → TeSys U n°2 → Preset Multiple Registers → Query → Data	0x0002	0x0202
Sub-Network → TeSys U n°3 → Preset Multiple Registers → Query → Data	0x0002	0x0204
Sub-Network → TeSys U n°4 → Preset Multiple Registers → Query → Data	0x0002	0x0206
Sub-Network → TeSys U n°5 → Preset Multiple Registers → Query → Data	0x0002	0x0208
Sub-Network → TeSys U n°6 → Preset Multiple Registers → Query → Data	0x0002	0x020A
Sub-Network → TeSys U n°7 → Preset Multiple Registers → Query → Data	0x0002	0x020C
Sub-Network → TeSys U n°8 → Preset Multiple Registers → Query → Data	0x0002	0x020E
Sub-Network → TeSys U n°1 → Read Holding Registers → Response → Data	0x0002	0x0000
Sub-Network → TeSys U n°2 → Read Holding Registers → Response → Data	0x0002	0x0002
Sub-Network → TeSys U n°3 → Read Holding Registers → Response → Data	0x0002	0x0004
Sub-Network → TeSys U n°4 → Read Holding Registers → Response → Data	0x0002	0x0006
Sub-Network → TeSys U n°5 → Read Holding Registers → Response → Data	0x0002	0x0008
Sub-Network → TeSys U n°6 → Read Holding Registers → Response → Data	0x0002	0x000A
Sub-Network → TeSys U n°7 → Read Holding Registers → Response → Data	0x0002	0x000C
Sub-Network → TeSys U n°8 → Read Holding Registers → Response → Data	0x0002	0x000E

N.B. The "Byte swap" field must *always* have the value "No swapping" if you wish to avoid inverting the LSB and MSB bytes of the data read or written using the Modbus commands.

ENGLISH

6. Advanced Implementation of the Gateway

- **Description of Services Assigned to Gateway** .....Section 4.2.8 ... Page 34
  ⇒ In the example provided, these services are identical to the ones in the standard configuration, for the following reasons:
  - The *periodic communications (inputs / outputs)* are configured so that the gateway input / output configuration (see above) is identical to the one in the standard configuration.
  - The *aperiodic communications* corresponding to the gateway's PKW service must be active, whether the gateway is configured according to profile FED C32 (in PL7 PRO) or profile FED C32 P (in AbcConf).
  - The *list of active slaves (LAS)* must also be active on the gateway.

- **Validating and Saving the FIPIO Network Configuration**.....Section 4.2.9 ... Page 35
  ⇒ These operations have no equivalent under AbcConf, as AbcConf can configure only one gateway at a time. The operations you can perform on a configuration are: Open and Save.
  ⇒ Under PL7 PRO, by contrast, you must always validate FED C32 P profile configuration and setting.

- **Assigning Symbols to the Gateway's FIPIO Objects** .....Section 4.2.10 ... Page 35
  ⇒ The FIPIO objects remain the same, except for the 30 configuration parameters (%KW\p.2.c\0.0 to %KW\p.2.c\0.0.29) and the 30 adjustment parameters (%MW\p.2.c\0.0.20 to %KW\p.2.c\0.0.49), which disappear, as their function is now taken over by AbcConf.

- **Checking the Operational Status of the Gateway**.....Section 4.2.11 ... Page 36
  ⇒ If you want to use the example "`LUFP1_FEDC32_Example.cfg`", run AbcConf, open the configuration file ("Open…" command in the "File" menu), transfer it to the gateway ("Download configuration to ABC-LUFP" command in the "File" menu) and close AbcConf.
  ⇒ Once the configuration has been downloaded to the gateway, and the PL7 PRO application transferred to the FIPIO master PLC, use PL7 PRO, in ONLINE mode, to check that the basic module `0 | FED C32` corresponding to the gateway is operational on the FIPIO network.
  ⇒ A "Hardware configuration fault" will occur if you downloaded an AbcConf configuration based on FED C32 P rather than FED C32, which is not the case for the aforementioned example file.

- **Modifying the Gateway Parameters in ONLINE Mode** .....Section 4.2.12 ... Page 37
  ⇒ This operation is now performed under AbcConf instead of under PL7 PRO. Downloading a new configuration to the gateway reinitialises the gateway, and in the process its FIPIO and Modbus communications.

- **Debugging and Using the Gateway Configuration** .....Section 4.2.13 ... Page 37
  ⇒ The diagnostics, inputs and outputs of the gateway are accessible, in ONLINE mode, via the debug window called up by double-clicking the basic module `0 | FED C32` corresponding to the LUFP1 gateway.

- **Developing a FIPIO Application**.....Section 4.2.14 ... Page 37
  ⇒ The PLC application example in Chapter 11 Appendix D: Sample Use under PL7 PRO, page 114, is fully compatible with the gateway when configured using profile FED C32. The only modification required is to change the gateway profile type (double-click the FIPIO connection point and select profile FED C32 instead of FED C32 P).

ENGLISH

7. Using ABC-LUFP Configurator

Each part of this chapter describes a separate step allowing users to personalise the gateway configuration according to their own particular needs, but ***only when it is configured using standard profile*** **FED C32** ***under PL7 PRO***. Each part introduces a basic operation, isolating it from the rest of the configuration and describing the operations to be carried out using AbcConf and how they affect the gateway's general behaviour.

Some of the operations can also be performed ***when the gateway is configured using standard profile*** **FED C32 P**. This possibility will be flagged by the symbol **FED C32 P** in the corresponding description.

In each case, the first two steps are required, as they allow you to establish the dialogue between the gateway and the PC software that enables you to configure it, i.e. AbcConf.

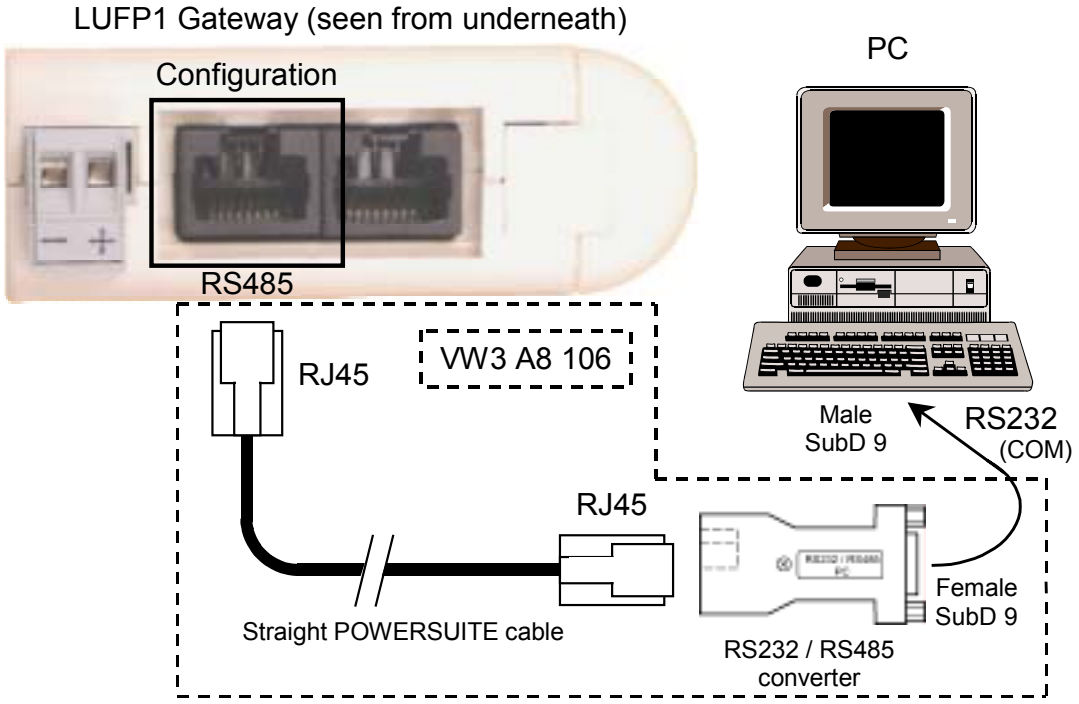
We strongly recommend that you read Chapter 4 Software Implementation of the Gateway, page 25, and, above all, Chapter 6 Advanced Implementation of the Gateway, page 56, as all the AbcConf operations described here assume that we are using the standard configuration (with 8 TeSys U motor starters) presented in the context of the two LUFP1 gateway implementation methods.

7.1. Connecting the Gateway to the Configuration PC

This step is required when setting up the gateway configuration application, AbcConf.

Connecting the gateway to one of the serial (COM) ports on a PC requires a straight PowerSuite cable and an RS232/RS485 converter. These two items are the same as those allowing dialogue with drives and soft start/soft stop units using the **PowerSuite** application and are both available from the catalogue (ref.: VW3 A8 106).

Be sure to use the "POWERSUITE" cable and the "RS232 / RS485 PC" converter. An "ATV28 before 09 / 2001" cable and an "ATV 58" converter are also supplied with these items, but they should not be used with the LUFP1 gateway.

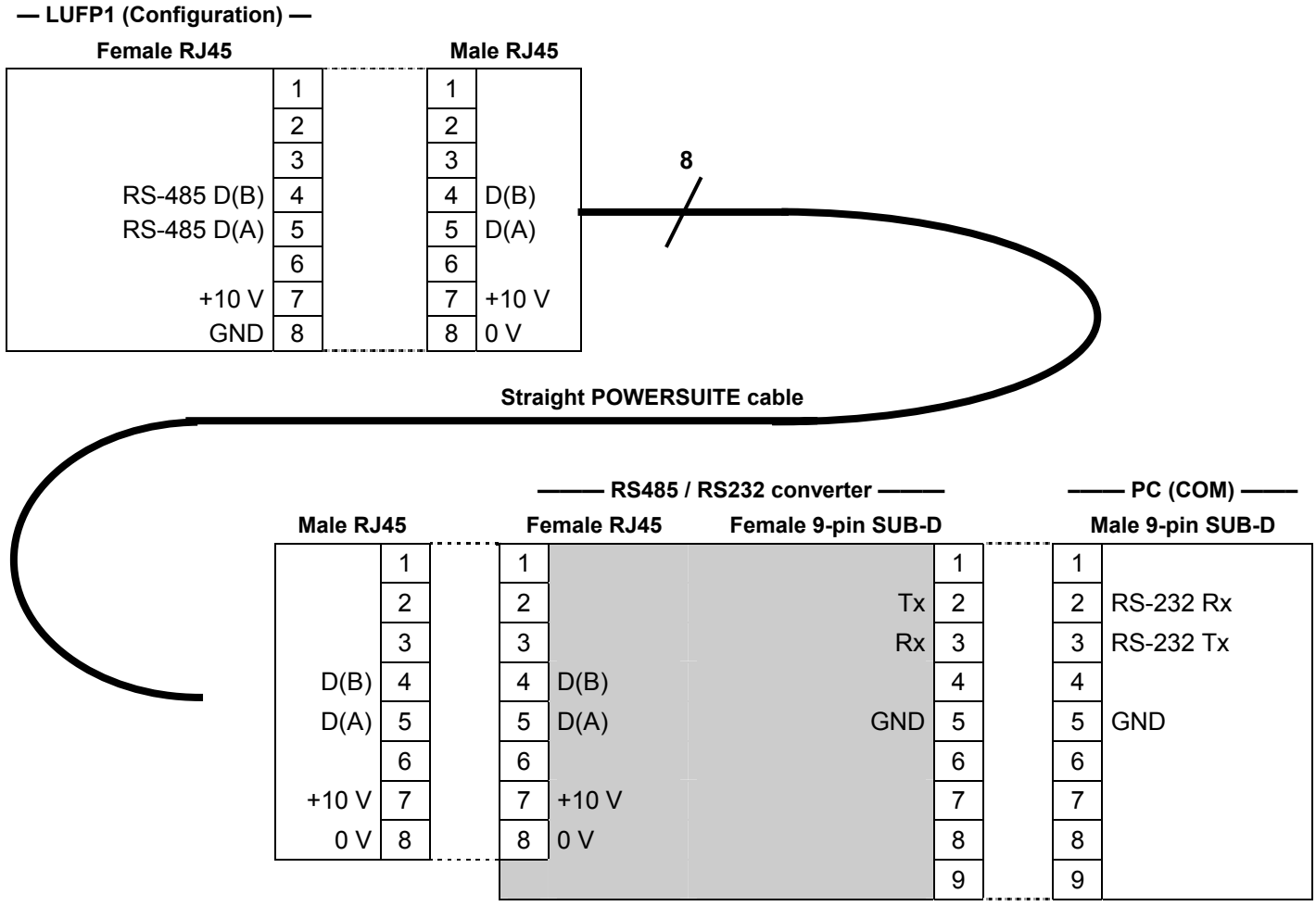


Once the gateway has been connected to a PC with the PowerSuite cable and the RS232/RS485 converter, you can change its configuration using "ABC-LUFP Configurator", more generally referred to as "AbcConf". This configuration tool also allows you to carry out a few diagnostics on the gateway.

ENGLISH

7. Using ABC-LUFP Configurator

7.1.1. Pin-Outs



N.B. The inversion of the Rx and Tx signals between the gateway and the PC is shown on the 9-pin Sub-D connectors, as beyond this junction, the RS-232 signals are replaced by the D(A) and D(B) polarisations of the RS-485 signals.

7.1.2. RS-232 Link Protocol

There is no need to configure the PC's COM port, as AbcConf uses a specific setup which replaces the one for the port being used. This replacement is temporary and is cancelled as soon as AbcConf stops using this serial port, i.e. when AbcConf is closed.

ENGLISH

7. Using ABC-LUFP Configurator

7.2. Installing AbcConf

The minimum system requirements for AbcConf are as follows:

- Processor.....Pentium 133 MHz
- Free hard disk space .....10 MB
- RAM..... 8 MB
- Operating system .....MS Windows 95 / 98 / ME / NT / 2000
- Browser.....MS Internet Explorer 4.01 SP1

The AbcConf installation program can be found on the CD LU9CD1. To install it, run "`ABC-LUFP_Setup.exe`" and follow the on-screen instructions.

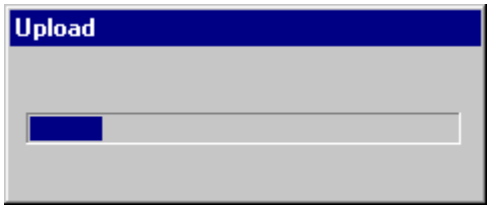
You can read about how to use AbcConf in a user manual entitled **AnyBus Communicator – User Manual** which is also on CD LU9CD1, under "`ABC_User_Manual.pdf`". We strongly recommend that you read this manual when using AbcConf: the present guide will limit itself to the AbcConf features relevant to a LUFP1 gateway implementation.

7.3. Retrieving the Gateway Configuration

Before you can make any changes to the gateway configuration, you will first need to retrieve its current configuration. If you already have this configuration on your hard disk, all you will need to do is open the file corresponding to this configuration.

Check that the gateway has a valid configuration and that it is working properly, i.e. that LED ❻ GATEWAY is flashing green.

In AbcConf, choose "Upload configuration from ABC-LUFP" from the "File" menu or click the  button in the AbcConf toolbar. The "Upload" window opens, containing a progress bar that tracks the advance of the gateway configuration uploading process. The window closes once the upload is completed.



This step is particularly important if you want to read details about the *current* contents of the gateway configuration rather than about a configuration in a "`.cfg`" file on a PC. You can then use this configuration as a template for any changes you wish to make subsequently, thus avoiding having to create one from scratch and reducing the potential risk of error.

Save this configuration to your hard disk so that it is always available. This will allow you to reconfigure the gateway "cleanly" should the configuration become invalid, if you were to download an invalid configuration, for example.

N.B. The standard configuration described in Chapter 6 Advanced Implementation of the Gateway (page 56) can be found on CD LU9CD1 under "`LUFP1_FEDC32_Example.cfg`".

FED C32 P You can use the gateway configuration retrieval command to check the configuration and settings that you made under PL7 PRO. This is because the gateway creates an equivalent configuration that can be viewed using the retrieve command.

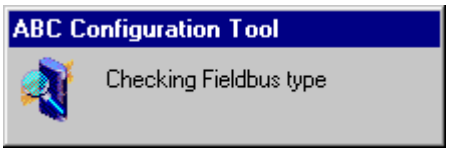
ENGLISH

7. Using ABC-LUFP Configurator

7.4. Transferring a Configuration to the Gateway

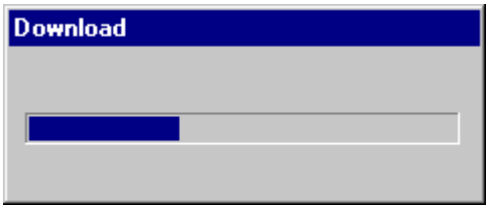
When using AbcConf, you can transfer the configuration you are editing to the gateway at any time.

Choose "Download configuration to ABC-LUFP" from the "File" menu or click the  button in the AbcConf toolbar.



AbcConf launches a check test of the gateway type. **During this test, the PC should not carry out any other operations, as this could lead to AbcConf apparently freezing up and slow down the PC's general operation for several minutes!** The test then continues and the PC returns to normal running speed.

Once this test has finished, the "Download" window opens containing a progress bar that tracks the transfer of the configuration to the gateway. Do not interrupt this operation, or you will have to start over again from the beginning.



Check that the transfer has been correctly carried out: LED ❻ GATEWAY should be flashing green.

If this LED is flashing red/green, save the configuration you were editing, open a file containing a valid configuration ("`LUFP1_FEDC32_Example.cfg`", for example), and transfer it to the gateway. This will restore it to a known initial state. You can then continue with the configuration you were transferring, and make any corrections which may be necessary.

7.4.1. Transferring the Internal Configuration (Profile FED C32 P)

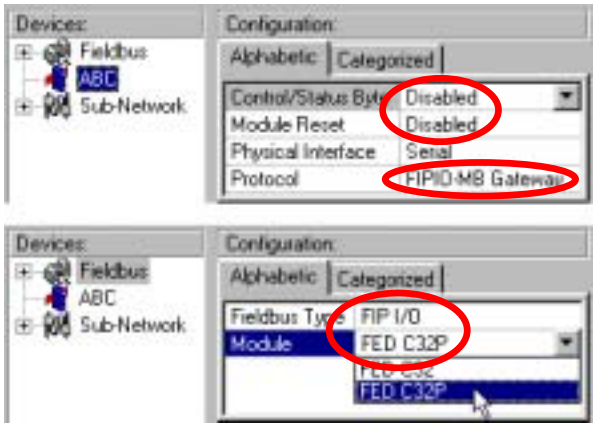
FED C32 P

When using the standard profile FED C32 P to set up the gateway under PL7 PRO, ***never*** use the command "Download configuration to ABC-LUFP" to transfer a configuration to the gateway.

The only exception to this rule involves the "Hardware configuration fault" (see Section 4.2.11 Checking the Operational Status of the Gateway, page 36).

In the latter instance, you should use the "New" command in the "File" menu:

- Select the protocol "FIPIO-MB Gateway", and the control / status byte and module reset "disabled" in the Abc options.
- Select the Fieldbus "FIPIO" and the profile "FEDC32P" in the network options.
- Do not add a Modbus command.
- Download the configuration to the gateway.



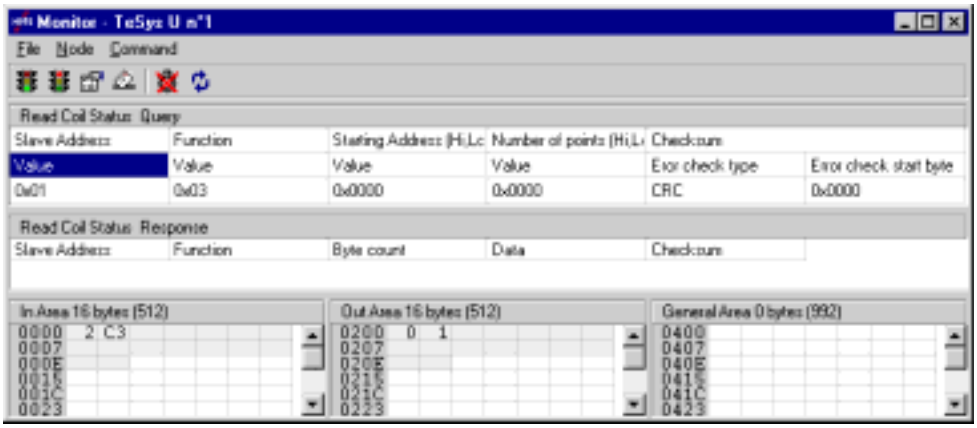
7. Using ABC-LUFP Configurator

7.5. Monitoring the Contents of the Gateway's Memory

One of the main commands that you will need to use when setting up the gateway is the command allowing you to read the contents of the gateway's memory and to display it in a window designed for this purpose. This will be particularly useful when you are working on your PLC applications and configurations. However, it only shows data from the "Data" and "Preset Data" fields configured in the "Query" and "Response" elements of a single Modbus slave.

As the gateway's LAS and PKW services are not located in the memory zones reserved for exchanges with the Modbus slaves, the input and output data of these services cannot be viewed in the same way. Only PL7 PRO and the FIPIO master can access such data!

To monitor the contents of the gateway's memory, start by selecting the node corresponding to the Modbus slave whose data you wish to view, then choose "Monitor" from the menu with the same name as the previously selected node. A monitoring window then appears.



The sample window above displays the contents of the memory exchanged, under the standard configuration, with TeSys U motor starter n°1.

In the upper pane you can choose a Modbus command, edit its contents, and send it to the Modbus network ("Command" menu). The response will then be displayed in the same pane. See Section 2.10 "Node Monitor" in the AbcConf user manual, entitled **AnyBus Communicator – User Manual**, for further information about how to use this window. The manual can be found on CD LU9CD1 under "`ABC_User_Manual.pdf`"".

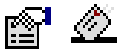
The lower pane allows you to view the contents of the gateway's memory, but only the bytes used in the query and response frames of the commands configured for the selected node.

In the window shown above, the data displayed correspond to the values at the memory locations designated by the "Data" fields in the commands configured for the node "TeSys U n°1", namely: "Read Holding Registers" and "Preset Multiple Registers".

N.B. The data exchanged with the previously selected Modbus slave are displayed MSB-first, i.e. in the order MSB / LSB (from left to right, in ascending order of memory address), provided that the "Byte Swap" option in the "Data" or "Preset Data" element of the corresponding Modbus command was set to "No swapping" (see Section 7.11.2.4 Configuring the Content of the Query Frame, page 87).

A brief description of the toolbar buttons in this window is given below:


Stop / Start communications with the selected node


Select / Send the Modbus command in the upper pane


Stop / Resume refreshing the data in the lower pane

ENGLISH

7. Using ABC-LUFP Configurator

FED C32 P Proceed by Retrieving the Gateway Configuration (Section 7.3, page 63) before Monitoring the Contents of the Gateway's Memory. This will ensure that the data displayed correspond to the inputs and outputs exchanged with the selected Modbus slave.

7.6. Creating a New Configuration

When you want to create a new configuration, you can either use an existing configuration as a starting-point—as is the case with most of the operations described in Chapter 7—or you can use a blank configuration.

For a blank configuration, use the "New" command in the "File" menu:

- Select the protocol "FIPIO-MB Gateway" in the Abc options;
- Select the fieldbus "FIPIO" and the profile "FEDC32" in the network options.

FED C32 P The creation of new configurations is performed *entirely* under PL7 PRO. Only transfer a new configuration to the gateway if it signals a "Hardware configuration fault" (see Section 7.4.1 Transferring the Internal Configuration (Profile FED C32 P), page 64).

7.7. Deleting a Modbus Slave

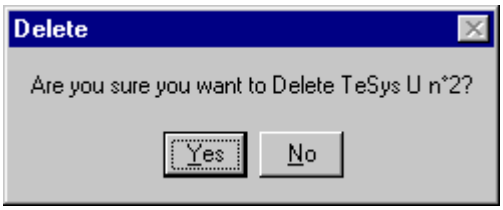
This step allows you, for instance, to free up a location on the downstream Modbus network, known as the "Sub-Network" in AbcConf, in order to replace one Modbus slave with another.

In fact the gateway's standard configuration already allows it to communicate with eight TeSys U motor starters, and the maximum number of Modbus slaves with which it is permitted to communicate is limited to eight.

If the gateway is used to manage exchanges on a Modbus network with fewer than eight TeSys U motor starters, it is preferable to delete the redundant TeSys U motor starters from the gateway configuration: the deterioration in performance caused by the absence of one or more TeSys U motor-starters is such that it is preferable to carry out this operation using AbcConf.

Procedure for deleting a Modbus slave

1) Select the node corresponding to the Modbus slave you wish to delete from the configuration. If this is the only node remaining in the configuration, you will not be able to delete it, as the downstream Modbus network must include at least one slave.
2) Right click on the icon or the name of this Modbus slave. A menu pops up underneath the mouse cursor.
   or
   In the AbcConf main menu, pull down the menu with the same name as the previously selected node.
3) In this menu, click "Delete". The following confirmation window then appears, asking you to either confirm that you want to delete the selected node ("TeSys U n°2" in this example) or cancel the operation.
4) If you confirm that you want to delete the node, the menu disappears, along with the previously selected node. Otherwise, the node will still be there when the window closes.



Keyboard shortcut: "Del" key.

7. Using ABC-LUFP Configurator

7.8. Adding a Modbus Slave

This feature allows you to add a Modbus slave of a different type to the other Modbus slaves in the configuration. On the other hand, if the slave is of the same type as one of the previously configured slaves, it is preferable to copy the old slave rather than create a new one.

An additional import/export feature also allows you to individually save the complete configuration of a Modbus slave, in order to have access to it in AbcConf, from any configuration and at any time.

These two features are only available provided that there are fewer than 8 Modbus slaves declared, which is not the case in the standard configuration, as it comprises 8 TeSys U motor starters.

Adding a new type of Modbus slave

Use one of the two methods described below:

a) Select “Sub-Network”, then choose “Add Node” from the “Sub-Network” menu. A new node is added after all the other configured nodes. By default, its name is “New Node”.

b) Select one of the nodes located under the “Sub-Network” element, then choose “Insert New Node” from the menu with the same name as the selected node. A new node is added just before the selected node. By default, its name is “New Node”.

All of the steps in configuring the new node are described in Section 7.10 Changing a Modbus Slave Configuration, page 78.

Copying a previously configured Modbus slave

Select the node corresponding to the slave whose configuration you want to copy, then choose “Copy” from the menu with the same name as the selected node. **Keyboard shortcut:** “Ctrl C”.

Then use one of the two methods described below:

a) Select “Sub-Network”, then choose “Paste” from the “Sub-Network” menu. A new node is added after all the other configured nodes. Its name and its whole configuration are identical to that of the node you copied. **Keyboard shortcut:** “Ctrl V”.

b) Select one of the “Sub-Network” nodes, then choose “Insert” from the menu with the same name as the selected node. A new node is added just before the one which is selected. Its name and its whole configuration are identical to that of the node you copied.

As the new node and the original node are identical in every way, you will need to change (1) the name of the node, (2) the address of the corresponding Modbus slave and (3) the location of the data exchanged between the gateway memory and the Modbus slave. All these operations are described in Section 7.10 Changing a Modbus Slave Configuration, page 78, and Section 7.11 Adding and Setting Up a Modbus Command, page 80.

Importing/exporting a Modbus slave configuration

With AbcConf you can independently save and load a node configuration on the downstream “Sub-Network”. This allows you, for example, to build up a library of Modbus slave templates, so that you can use them in any configuration.

To save the configuration of a Modbus slave, select the node it corresponds to, then choose “Save Node” from the menu with the same name as the selected node. A dialog box will then appear asking you to save the configuration (export in XML format).

7. Using ABC-LUFP Configurator

To insert a node using the XML file containing a Modbus slave configuration as a template, use one of the two methods described below:

a) Select “Sub-Network”, then choose “Load Node” from the “Sub-Network” menu. A dialog box asks you to choose a file containing a Modbus slave configuration (import in XML format). A new node is added after all the other configured nodes. Its name and its whole configuration are identical to those of the Modbus slave as configured when last saved.

b) Select one of the “Sub-Network” nodes, then choose “Insert from File” from the menu with the same name as the selected node. A new node is added just before the selected node. Its name and its whole configuration are identical to those of the Modbus slave as configured when last saved.

You will then need to change (1) the name of the node, (2) the address of the corresponding Modbus slave and (3) the location of the data exchanged between the gateway memory and the Modbus slave. All these operations are described in Section 7.10 Changing a Modbus Slave Configuration, page 78, and Section 7.11 Adding and Setting Up a Modbus Command, page 80.

7.9. Modifying Periodic Data Exchanged with Modbus Slaves

This involves replacing, adding or deleting periodic data exchanged with one of the Modbus slaves. For each of these operations, we shall take the standard configuration as described in Chapter 6 Advanced Implementation of the Gateway (page 56) as the starting-point: any changes previously made will have been cancelled at the start of each operation. Each operation is illustrated below by a specific example.

Always remember to save the changes you have made, and to transfer the whole configuration to the gateway. This will allow you to check that the configuration is valid.

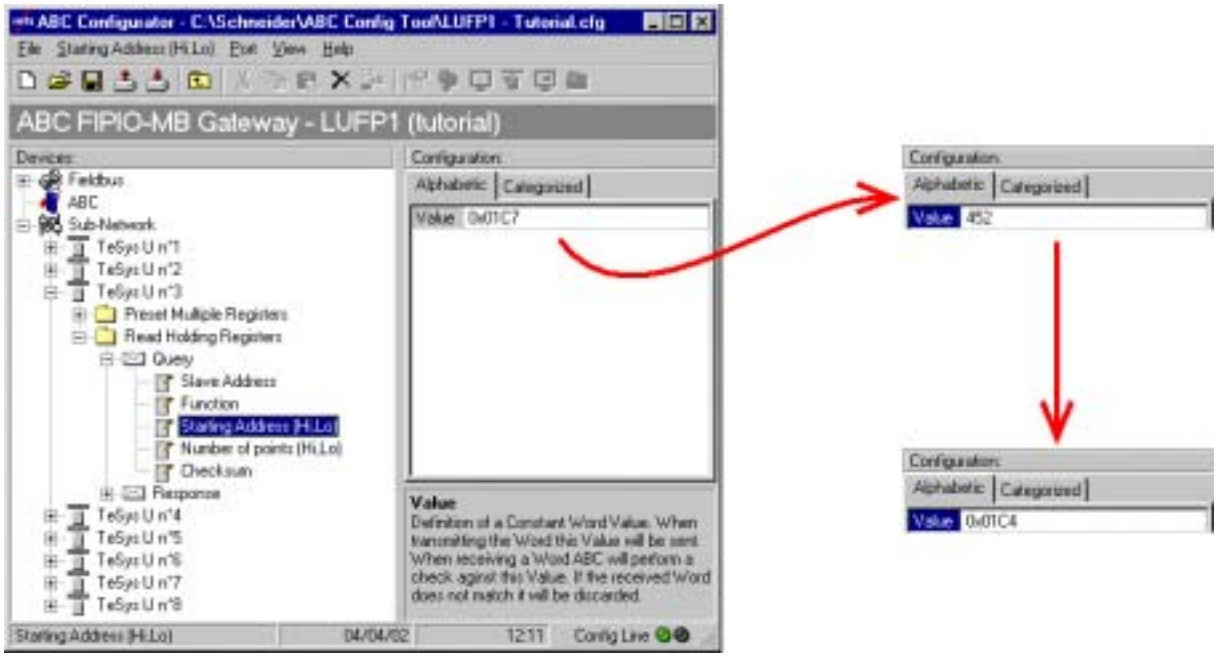
7.9.1. Replacing a Periodic Input

E.g. TeSys U Motor starter n°3: we want to replace the “TeSys U Status Register” control (address 455 = 16#01C7) with the “1st Fault Register” control (address 452 = 16#01C4).

The operation is a very simple one and consists merely of changing the value of the “Starting Address (Hi, Lo)” element in the “Query” of the “Read Holding Registers” command (Modbus command for reading values off multiple registers).

Select this element, then change its value as shown at the top of the next page. You can enter the address of the parameter in decimal format: AbcConf will automatically convert it to hexadecimal.

7. Using ABC-LUFP Configurator



This operation in no way alters the contents of the gateway memory, as we do not need to change the values of the "Data length" and "Data location" fields in the "Data" element of the "Response" to the aforementioned command; so no additional operations are required in AbcConf.

However, the FIPIO master PLC application will have to take account of the change in the nature of the corresponding input. In Section 10.2.1 Input Data Memory Area, page 112, the description of the word located at address 16#0004 becomes "Value of the 1st fault register of motor starter ③". This word corresponds to PLC input word %IW\p.2.c\0.0.2 (see Section 4.2.7 Configuring Gateway Inputs / Outputs, page 32, Section 5.1.1 Standard Configuration with 8 TeSys U Motor Starters, page 39, and Section 5.1.3 Configuration under AbcConf with Profile FED C32, page 41).

ENGLISH

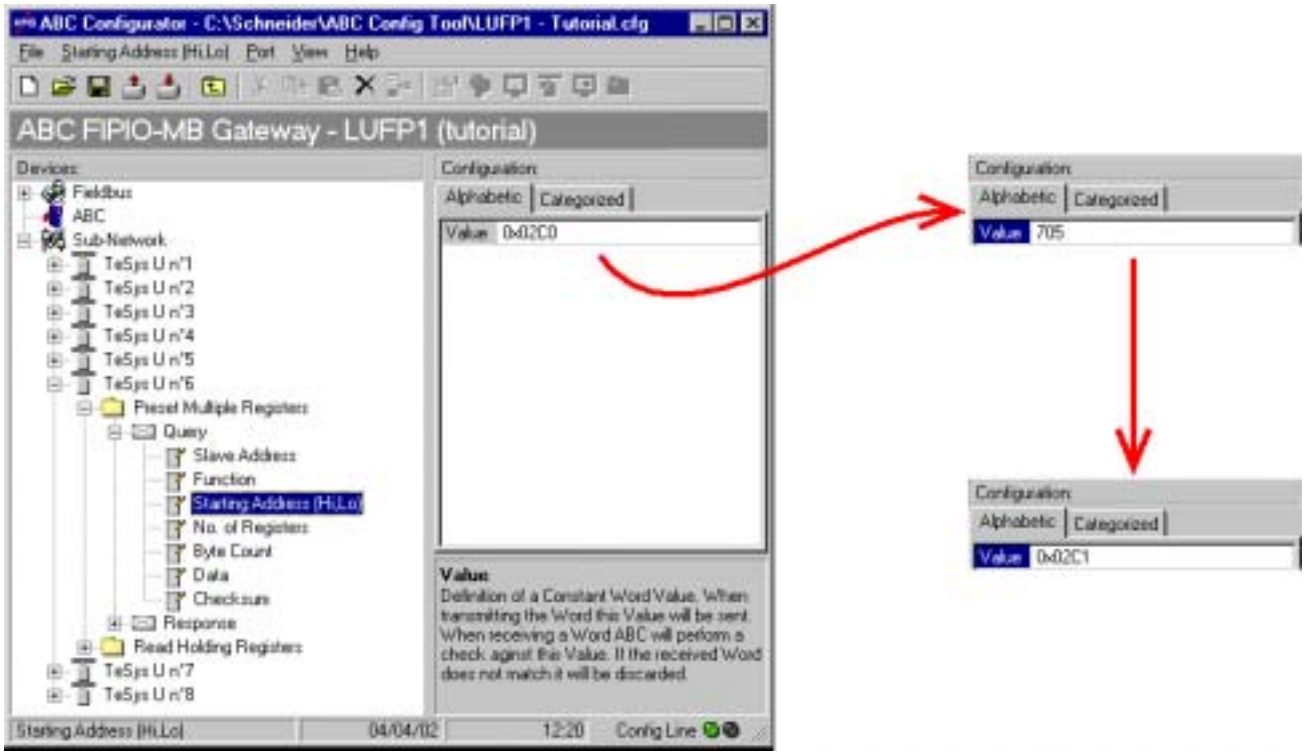
7.9.2. Replacing a Periodic Output

E.g. TeSys U Motor starter n°6: we want to replace the "Command Register" command (address 704 = 16#02C0) with the "2nd Command Register" command (address 705 = 16#02C1).

This operation consists in changing the value of "Starting Address (Hi,Lo)" in the "Query" and the value of "Starting Address" in the "Response" to the "Preset Multiple Registers" command (Modbus command for writing values to multiple registers).

Select "Starting Address (Hi,Lo)" in the "Query" and modify its value as shown at the top of the next page. You can enter the address of the parameter in decimal format: AbcConf will automatically convert it to hexadecimal. **Do the same for the "Starting Address" element of the "Response",** because the gateway checks the value of this field when it receives each Modbus response. If the value does not correspond to that of the query, the gateway will ignore the response.

7. Using ABC-LUFP Configurator



This operation in no way alters the contents of the gateway memory, as we do not need to change the values of the "Data length" and "Data location" fields in the "Data" element of the "Query". No additional operations are required in AbcConf.

However, the FIPIO master PLC application will have to take account of the change in the nature of the corresponding output. In Section 10.2.2 Output Data Memory Area, page 113, the description of the word located at address 16#020A becomes "Value of the 2nd command register of motor starter ⑥". This word corresponds to the PLC output word %QW\p.2.c\0.0.5 (see Section 4.2.7 Configuring Gateway Inputs / Outputs, page 32, Section 5.1.1 Standard Configuration with 8 TeSys U Motor Starters, page 39, and Section 5.1.3 Configuration under AbcConf with Profile FED C32, page 41).

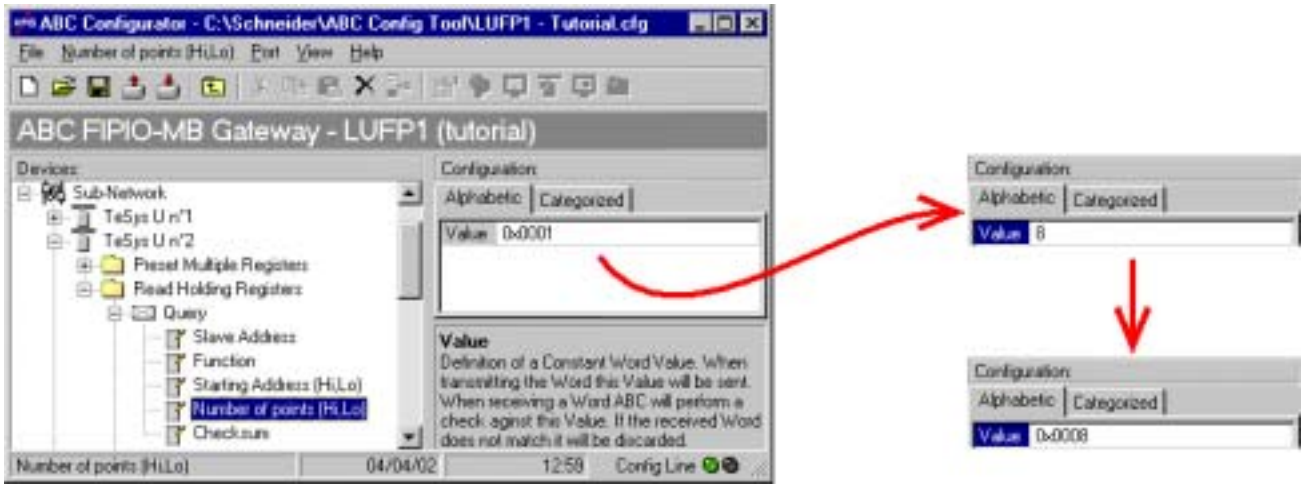
7.9.3. Increasing the Number of Periodic Inputs

E.g. TeSys U motor starter n°2": we want to extend the controls on this motor starter, starting with the currently controlled register—"TeSys U Status Register" (address 455 = 16#01C7)—and expanding as far as the "Reserved: 2nd Warning Register" (address 462 = 16#01CE). This will bring the number of registers controlled from 1 up to 8.

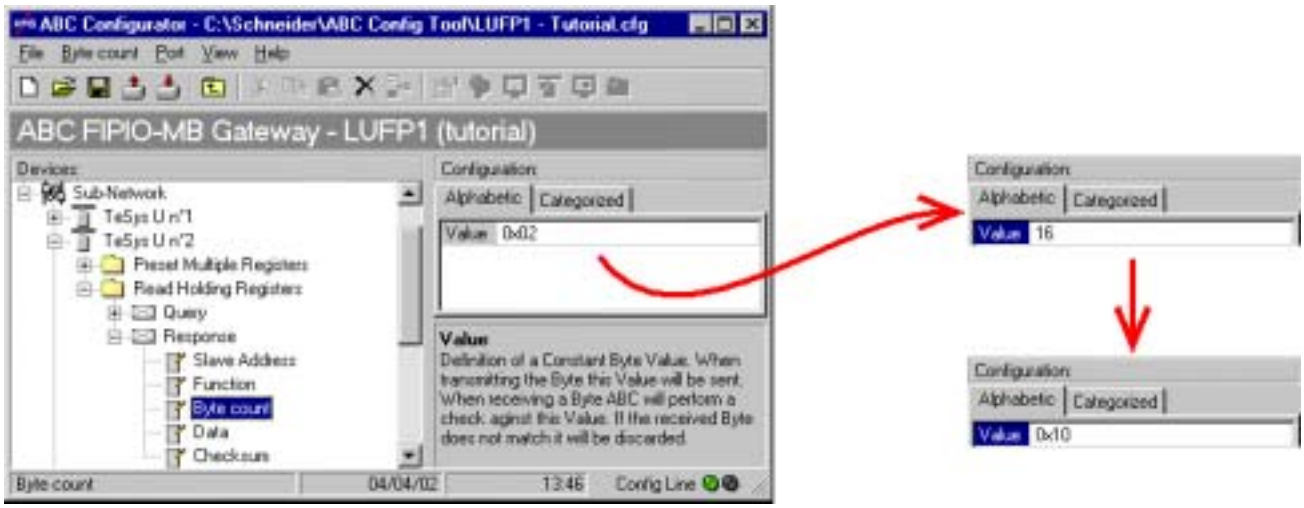
The necessary operations are described below:

1) Change the number of registers controlled: This step consists in modifying the value of the element "Number of points (Hi, Lo)" in the "Query" of the "Read Holding Registers" command (Modbus command for reading the values of multiple registers). Select this element, then change its value as shown at the top of the next page. AbcConf will automatically convert any value entered in decimal to hexadecimal.

7. Using ABC-LUFP Configurator



2) Change the number of data bytes in the Modbus response: The number of bytes read from the memory of TeSys U motor starter n°2 increases from 2 to 16, as the number of registers controlled has increased from 1 to 8. Select the "Byte count" element in the "Response" and change its value as shown below. AbcConf will automatically convert any value entered in decimal to hexadecimal.

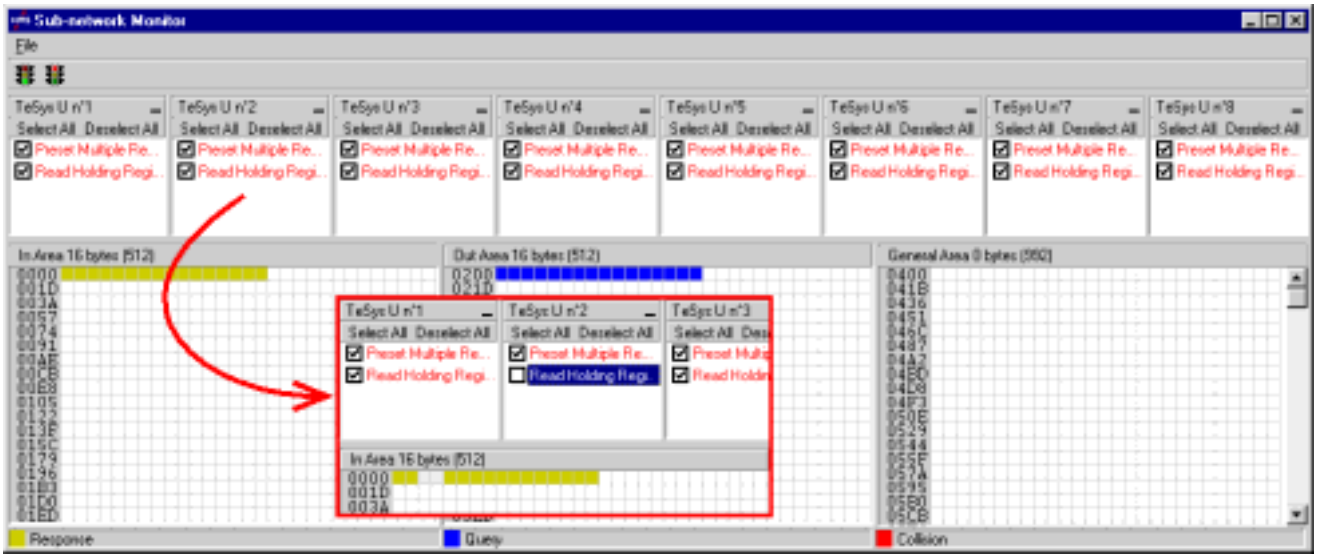


3) Change the location of the Modbus data received in the gateway memory: As the number of bytes read (see previous step) has increased from 2 to 16, the Modbus data received must be placed at a different location in the gateway memory.

   If you are unsure how much of the gateway's memory is currently in use, select "Sub-Network" and choose "Monitor" from the "Sub-Network" menu. The following window appears, allowing you to see how much of the gateway's memory is occupied.

ENGLISH

7. Using ABC-LUFP Configurator



To see which memory locations are occupied by data from the command we are interested in, simply uncheck the box corresponding to the “Read Holding Registers” command in the “TeSys U n°2” node, as shown above. This reveals that the Modbus data received in response to this command occupy 2 bytes, starting at 16#0002.

Only memory locations 16#0000 to 16#0033 can be used (max. size of input data = 52 bytes).

Using any other address in the gateway’s input data memory area (16#0034 to 16#01FF) will throw the gateway into ***non-operational*** mode, signalled by the sudden red-green flashing of LED ❻ GATEWAY. LEDs ❶, ❷, ❸ and ❹ will also flash, at a frequency of 2 Hz.

You can ignore the size indications on the graphics in this window (“In Area 16 bytes” and “Out Area 16 bytes”), as the gateway has a fixed size for inputs and outputs alike.

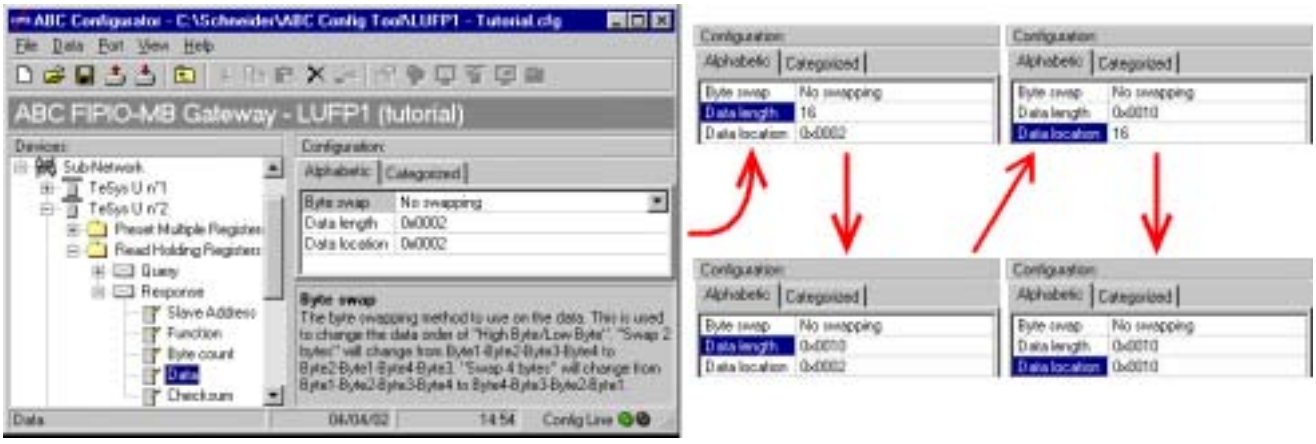
If you want to memorise the 16 bytes of Modbus data which will be received by the gateway for this command, once the changes have been made, you will have to either shift all the other input data by 14 bytes—a painstaking operation—or else change the memory location of the block of data received. In the example described here, we will be using the second solution, although the first solution is actually preferable, in principle, as it avoids leaving any “holes” in the gateway’s memory. This has no impact on the FIPIO exchanges, however, as the size of transfers over the gateway is set at 32 input words and 32 output words by the use of the standard profile FED C32 (P).

We will be placing the 16 bytes of data starting at address 16#0010 (16 in decimal), i.e. directly after the input data from the standard configuration. The 2 bytes at 16#0002 and 16#0003 become “free memory locations”.

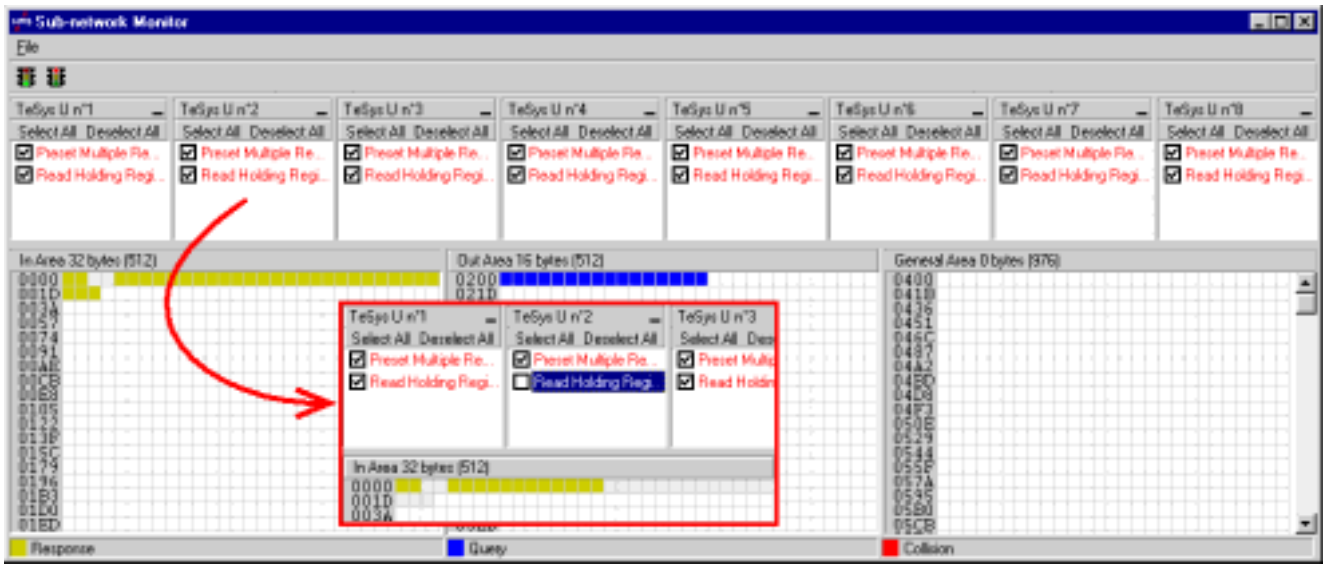
Close the “Sub-network Monitor” window, then, once you are back in the main AbcConf window, select the “Data length” and “Data location” fields respectively in the “Data” element of the “Response” and change their values as shown at the top of the next page. AbcConf will automatically convert any value entered in decimal to hexadecimal.

ENGLISH

7. Using ABC-LUFP Configurator



To check that these changes have been integrated into the configuration, choose the "Monitor" command again in the "Sub-Network" menu:



4) Download this configuration to the gateway: See Section 7.4 Transferring a Configuration to the Gateway, page 64. Check that the configuration is valid (LED ❻ GATEWAY flashing green).

5) Save the configuration to your PC's hard disk.

6) Use the FIPIO master PLC inputs: It only remains to assign symbols and use the inputs corresponding to the new Modbus data read on motor starter n°2.

   This gives us the mapping shown on the next page, derived from the one used for the standard gateway configuration. Changes in relation to the standard configuration are flagged by a grey background (except for the reserved word %IW\p.2.c\0.0.26).

ENGLISH

Service	PLC input	Description	
		Bit 15.....Bit 8	Bit 7.....Bit 0
Periodic communications — Control of TeSys U motor starters	%IW\p.2.c\0.0	Value of status register for motor starter ①	
	%IW\p.2.c\0.0.1	Free memory location	
	%IW\p.2.c\0.0.2	Value of status register for motor starter ③	
	%IW\p.2.c\0.0.3	Value of status register for motor starter ④	
	%IW\p.2.c\0.0.4	Value of status register for motor starter ⑤	
	%IW\p.2.c\0.0.5	Value of status register for motor starter ⑥	
	%IW\p.2.c\0.0.6	Value of status register for motor starter ⑦	
	%IW\p.2.c\0.0.7	Value of status register for motor starter ⑧	
Periodic communications — Control of TeSys U motor starter ②	%IW\p.2.c\0.0.8	Value of "TeSys U Status Register"	
	%IW\p.2.c\0.0.9	Value of "Complementary Status Register"	
	%IW\p.2.c\0.0.10	Value of "K7 Status Register"	
	%IW\p.2.c\0.0.11	Value of "K7 Status Register 2 (free format)"	
	%IW\p.2.c\0.0.12	Value of "K7 Status Register 3 (free format)"	
	%IW\p.2.c\0.0.13	Value of "Warning Number"	
	%IW\p.2.c\0.0.14	Value of "Warning Register"	
	%IW\p.2.c\0.0.15	Value of "Reserved: 2nd Warning Register"	
Periodic communications — Free locations	%IW\p.2.c\0.0.16 :::::::::: %IW\p.2.c\0.0.25	Not used (10 words)	
————	%IW\p.2.c\0.0.26	Reserved (1 word)	
List of active slaves (LAS) service	%IW\p.2.c\0.0.27	List of active Modbus slaves	
Aperiodic communications — Indexed periodic variables (PKW) service – RESPONSE	%IW\p.2.c\0.0.28	PKE – Address of the datum written / read	
	%IW\p.2.c\0.0.29	DN – Address of the responding slave	R/W/N – Read / Write OK / Error
	%IW\p.2.c\0.0.30	PWE – Value written / read (1st word)	
	%IW\p.2.c\0.0.31	PWE – Value written / read (2nd word)	

ENGLISH

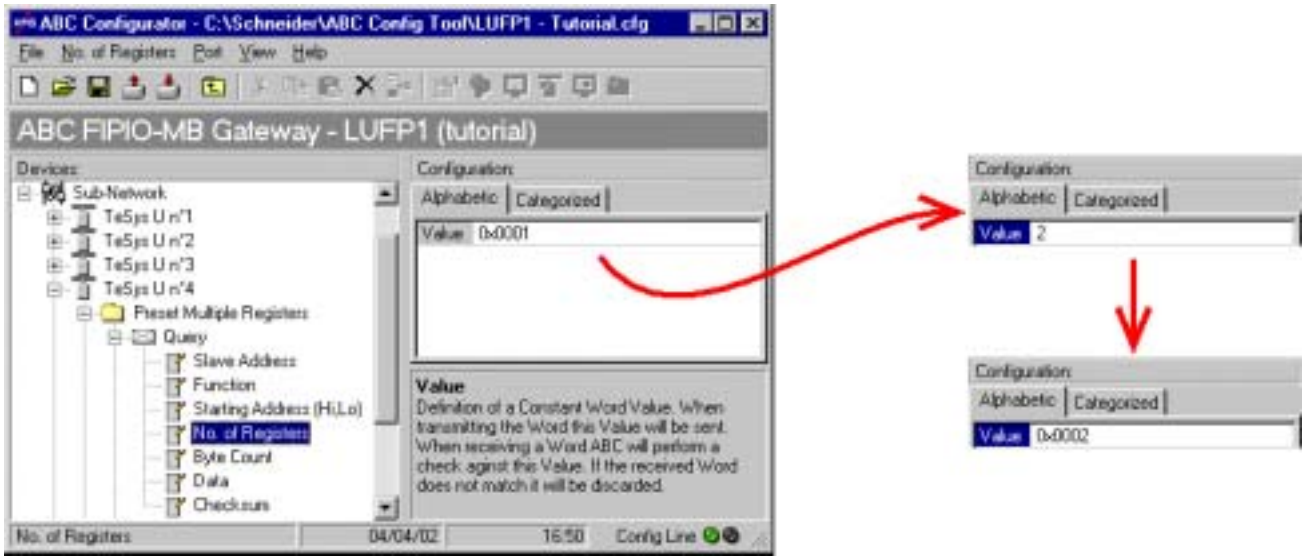
7.9.4. Increasing the Number of Periodic Outputs

E.g. TeSys U motor starter n°4: we want to extend the commands on this motor starter, retaining the currently commanded register, "Command Register" (address 704 = 16#02C0), and adding the next register, "2nd Command Register" (address 705 = 16#02C1). This will bring the number of registers commanded from 1 up to 2.

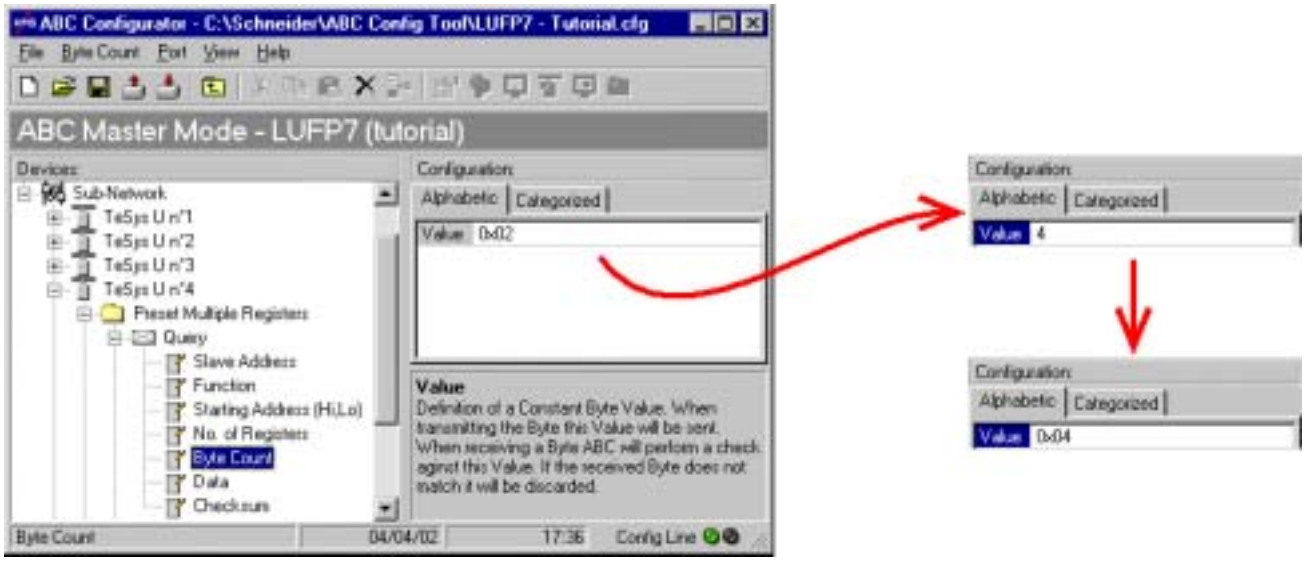
The necessary operations are described below:

1) Change the number of registers commanded: This step consists in modifying the value of the element "No. of Registers" in both the "Query" and "Response" of the "Preset Multiple Registers" command (Modbus command for writing values to multiple registers). Start by selecting the element "Starting Address (Hi,Lo)" in the "Query": change its value as shown at the top of the next page. AbcConf will automatically convert any value entered in decimal to hexadecimal. **Do the same for the "Starting Address" element of the "Response",** because the gateway checks the value of this field when it receives each Modbus response. If the value does not correspond to that of the query, the gateway will ignore the response.

7. Using ABC-LUFP Configurator



2) Change the number of data bytes in the Modbus query: The number of bytes written to the memory of TeSys U motor starter n°4 increases from 2 to 4, as the number of registers commanded has increased from 1 to 2. Select the "Byte count" element in the "Query" and change its value as shown below. AbcConf will automatically convert any value entered in decimal to hexadecimal.

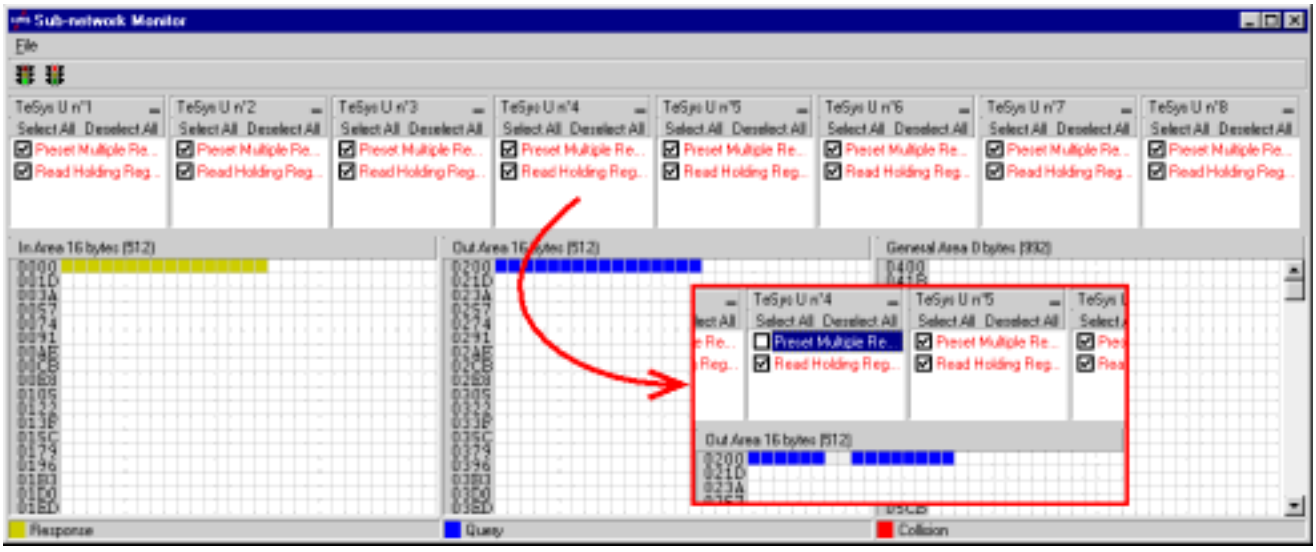


3) Change the location of the Modbus data transmitted to the gateway memory: As the number of bytes transmitted (see previous step) has increased from 2 to 4, the Modbus data to be sent to TeSys U motor starter n°4 must be placed at a different location in the gateway memory.

   If you are unsure how much of the gateway's memory is currently in use, select "Sub-Network" and choose "Monitor" from the "Sub-Network" menu. The following window appears, allowing you to see how much of the gateway's memory is occupied.

ENGLISH

7. Using ABC-LUFP Configurator



To see which memory locations are occupied by data from the command we are interested in, simply uncheck the box corresponding to the “Preset Multiple Registers” command in the “TeSys U n°4” node, as shown above. This reveals that the Modbus data transmitted with the query for the command occupy 2 bytes, starting at 16#0206.

Only memory locations 16#0200 to 16#0233 can be used (max. size of input data = 52 bytes).

Using any other address in the gateway’s output data memory area (16#0234 to 16#03FF) will throw the gateway into ***non-operational*** mode, signalled by the sudden red-green flashing of LED ❻ GATEWAY. LEDs ❶, ❷, ❸ and ❹ will also flash, at a frequency of 2 Hz.

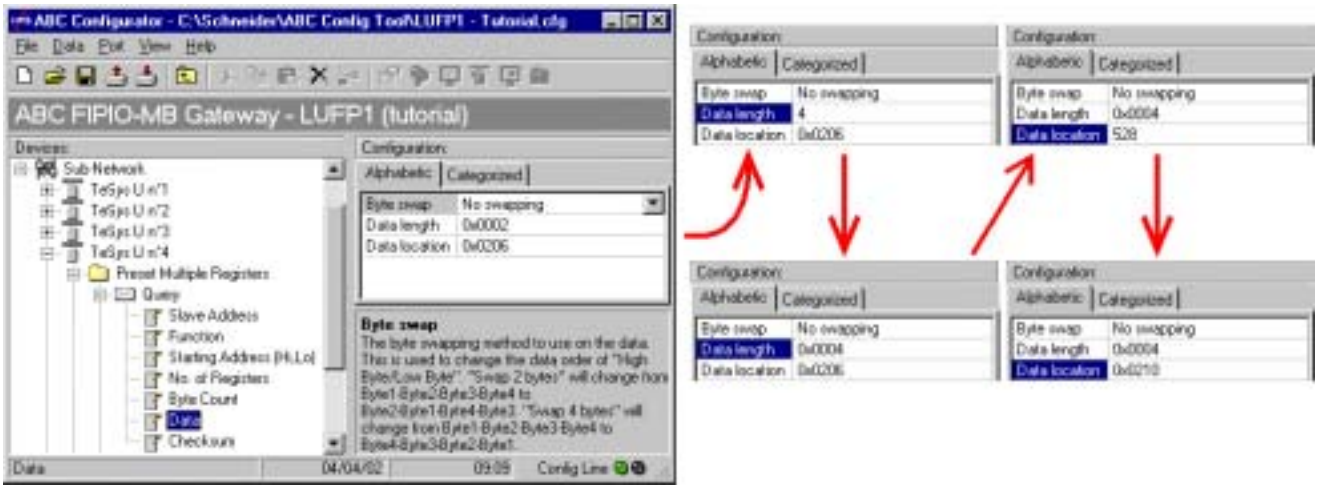
You can ignore the size indications on the graphics in this window (“In Area 16 bytes” and “Out Area 16 bytes”), as the gateway has a fixed size for inputs and outputs alike.

If you want to memorise the 4 bytes of Modbus data which will be transmitted by the gateway for this command, once the changes have been made, you will have to either shift all the other transmitted data by 2 bytes—a painstaking operation—or else change the memory location of the block of data transmitted. In the example described here, we will be using the second solution, although the first solution is actually preferable, in principle, as it avoids leaving any “holes” in the gateway’s memory. This has no impact on the FIPIO exchanges, however, as the size of transfers over the gateway is set at 32 input words and 32 output words by the use of the standard profile FED C32 (P).

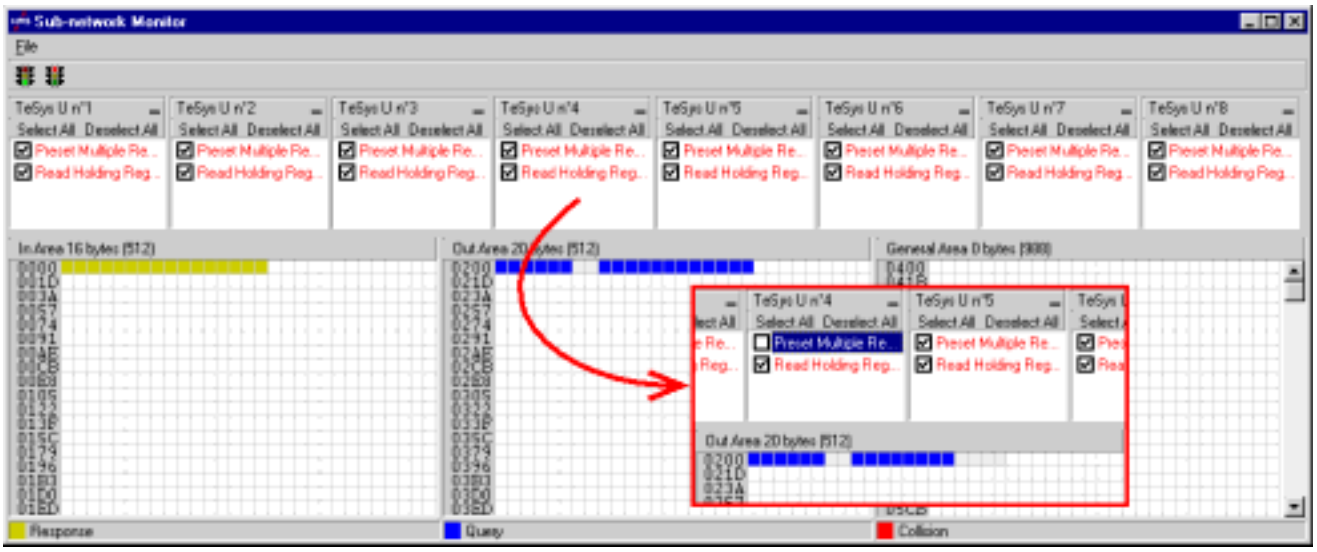
We will be placing the 4 bytes of data starting at address 16#0210 (528 in decimal), i.e. directly after the output data for the standard configuration. The 2 bytes at 16#0206 and 16#0207 become “free memory locations”.

Close the “Sub-network Monitor” window, then, once you are back in the main AbcConf window, select the “Data length” and “Data location” fields respectively in the “Data” element of the “Query” and change their values as shown at the top of the next page. AbcConf will automatically convert any value entered in decimal to hexadecimal.

7. Using ABC-LUFP Configurator



To check that these changes have been integrated into the configuration, choose the "Monitor" command again in the "Sub-Network" menu:



4) Download this configuration to the gateway: See Section 7.4 Transferring a Configuration to the Gateway, page 64. Check that the configuration is valid (LED ❻ GATEWAY flashing green).

5) Save the configuration to your PC's hard disk.

6) Use the FIPIO master PLC outputs: It only remains to assign symbols and use the outputs corresponding to the new Modbus data sent to motor starter n°4.

   This gives us the mapping shown on the next page, derived from the one used for the standard gateway configuration. Changes in relation to the standard configuration are flagged by a grey background (except for the two reserved words %QW\p.2.c\0.0.26 and %QW\p.2.c\0.0.27).

ENGLISH

Service	PLC output	Description	
		Bit 15.....Bit 8	Bit 7.....Bit 0
Periodic communications — Command of TeSys U motor starters	%QW\p.2.c\0.0	Value of command register for motor starter ①	
	%QW\p.2.c\0.0.1	Value of command register for motor starter ②	
	%QW\p.2.c\0.0.2	Value of command register for motor starter ③	
	%QW\p.2.c\0.0.3	Free memory location	
	%QW\p.2.c\0.0.4	Value of command register for motor starter ⑤	
	%QW\p.2.c\0.0.5	Value of command register for motor starter ⑥	
	%QW\p.2.c\0.0.6	Value of command register for motor starter ⑦	
	%QW\p.2.c\0.0.7	Value of command register for motor starter ⑧	
Periodic communications: — Command of TeSys U motor starter ④	%QW\p.2.c\0.0.8	Value of "Command Register"	
	%QW\p.2.c\0.0.9	Value of "2nd Command Register"	
Periodic communications — Free locations	%QW\p.2.c\0.0.10 ::::::::: %QW\p.2.c\0.0.25	Not used (16 words)	
_____	%QW\p.2.c\0.0.26 %QW\p.2.c\0.0.27	Reserved (2 words)	
Aperiodic communications — Indexed periodic variables (PKW) service – COMMAND	%QW\p.2.c\0.0.28	PKE – Address of the datum to be read / written	
	%QW\p.2.c\0.0.29	DN – Address of the queried slave	R/W – Read / Write command
	%QW\p.2.c\0.0.30	PWE – Value to be written (1st word)	
	%QW\p.2.c\0.0.31	PWE – Value to be written (2nd word)	

ENGLISH

7.10. Changing a Modbus Slave Configuration

Configuring a Modbus slave itself remains very simple because it only involves the name and the Modbus address of the node to which it corresponds. By contrast, configuring Modbus commands is a more complex business and is discussed in a separate section (see Section 7.11 Adding and Setting Up a Modbus Command, page 80).

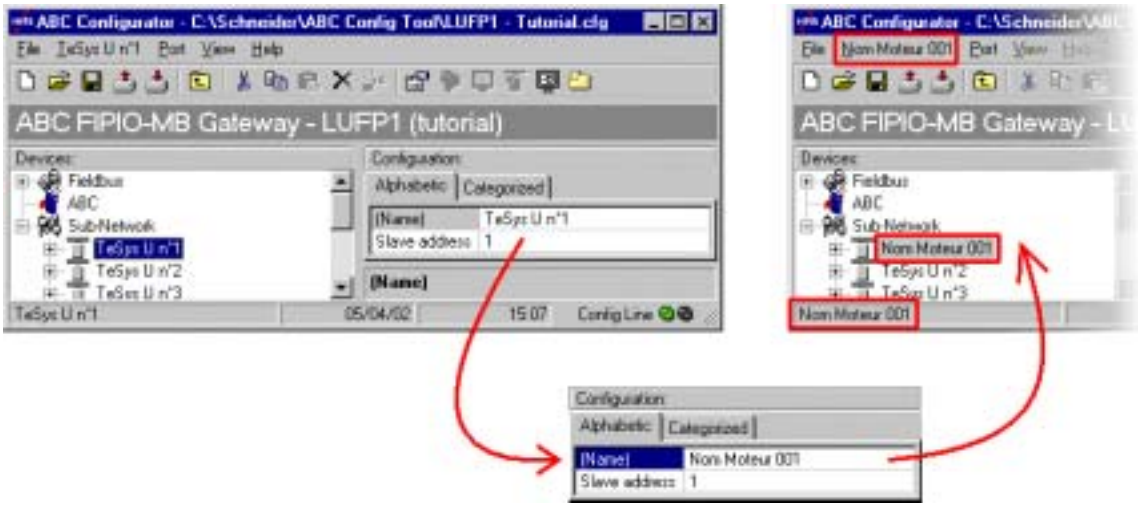
You will need to change the configuration of a Modbus slave when you add a new Modbus device (see Section 7.8 Adding a Modbus Slave, page 67), using any method.

Changing the name of the node which corresponds to a Modbus slave helps to distinguish it from the other nodes, for instance when its Modbus command configuration has been changed.

7.10.1. Changing the Name of a Modbus Slave

To carry out this operation, simply select the node which corresponds to the Modbus slave involved (under "Devices:"), click the current name (value of the "(Name)" field in the "Configuration:" pane), and change it. Once you confirm the new name ("Enter" key or click outside the name input field), it becomes effective in AbcConf, and the name of the node will be automatically updated in the "Devices:" pane. An example is given at the top of the next page. The three red frames shown in this example show the effects of the modification.

7. Using ABC-LUFP Configurator



7.10.2. Changing the Address of a Modbus Slave

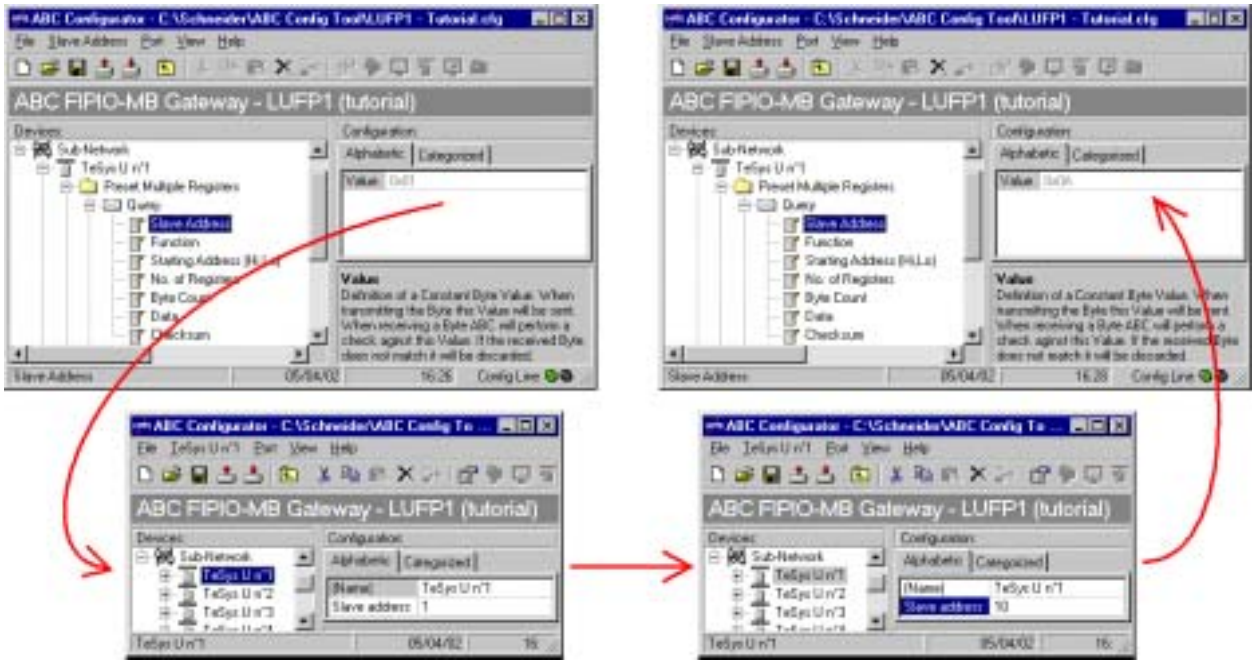
To carry out this operation, simply select the node which corresponds to the Modbus slave involved ("Devices:" pane), click on the value of the current address (value of the "Slave address" field in the "Configuration:" pane), and change it.

Reminder: The address of a Modbus slave must be between 1 and 247.

If you use Modbus slaves from the *Schneider Electric* Speed Variation range, such as Altistarts or Altivars, do not configure ANY slaves at addresses 65, 126 or 127 on the same Modbus network as the slaves in question, as these addresses are reserved when using these products.

ENGLISH

When you confirm the new address ("Enter" key or click outside the Modbus slave address input field), it becomes effective in AbcConf, and the values of the "Slave Address" elements in the queries and responses of the Modbus commands for the selected node are automatically updated. In the example that follows, a single "Slave Address" element is updated:



7. Using ABC-LUFP Configurator

7.11. Adding and Setting Up a Modbus Command

7.11.1. With TeSys U Motor Starters

With TeSys U motor starters, the main point of adding a Modbus command is to enable you to command or control additional registers, without having to change the elements in the standard configuration. The periodic and aperiodic communication services therefore operate in the same way as for the standard configuration, unlike the various operations described in Section 7.9 Modifying Periodic Data Exchanged with Modbus Slave, page 68.

Instead of adding a command and configuring it from scratch, it is easier to copy one of the two default commands for TeSys U motor starters—"Read Holding Registers" (read / control) or "Preset Multiple Registers" (write / command)—and paste it into the list of Modbus commands for the appropriate node.

To copy a ready-configured Modbus command, select it, then choose "Copy" from the menu with the same name as the selected command. **Keyboard shortcut:** "`Ctrl C`". Then proceed with either of the following methods:

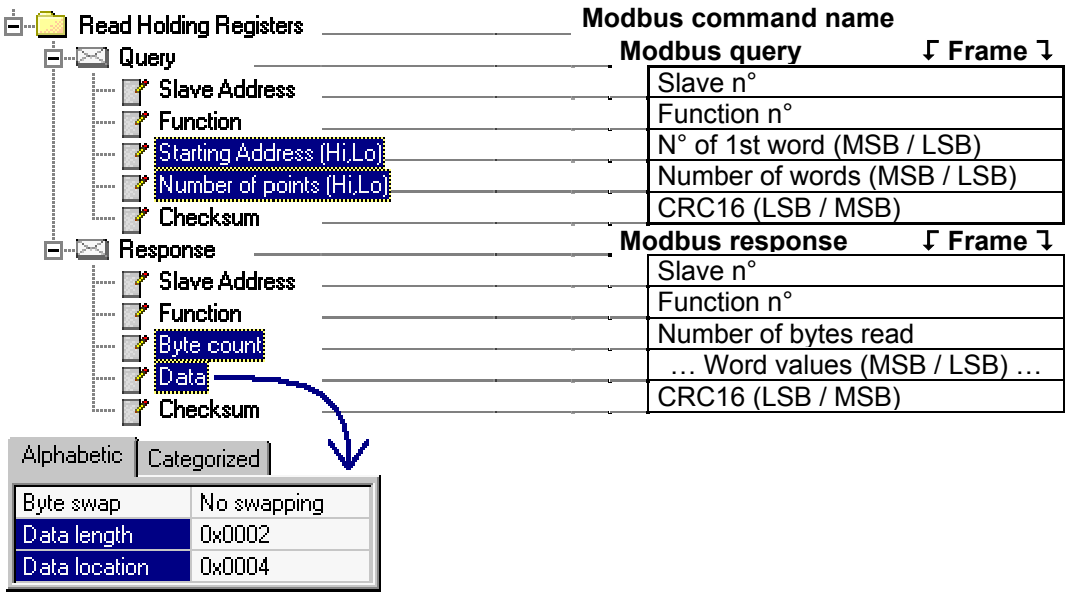
a) Select the node corresponding to the Modbus slave you want to add the command to (e.g. "TeSys U n°4"), then choose "Paste" from the menu with the same name as the selected node. A new command is added after all the other commands configured for this node. Its configuration is totally identical to that of the previously copied command. **Keyboard shortcut:** "`Ctrl V`".

or:

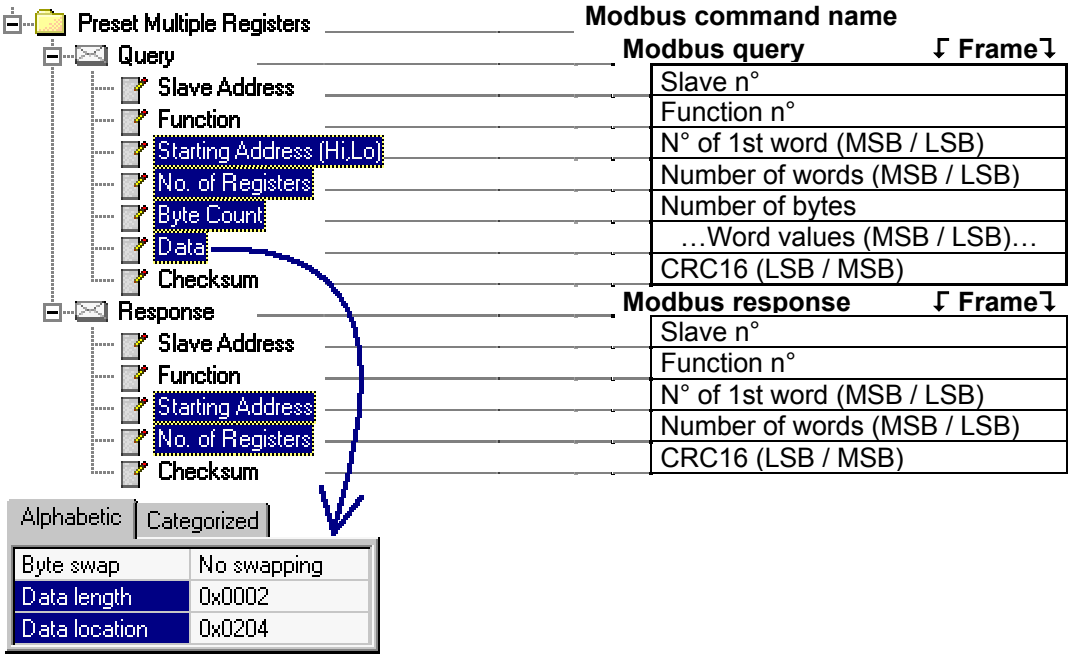
b) Select one of the commands on the node concerned, then choose "Insert" from the menu with the same name as the selected command. A new command is added just before the one selected. Its configuration is totally identical to that of the previously copied command.

ENGLISH

As the new Modbus command and the original Modbus command are identical, you will need to make changes to the fields highlighted in blue in one of the two diagrams below, depending on whether it is a "Read Holding Registers" command or a "Preset Multiple Registers" command (see Section 7.9 Modifying Periodic Data Exchanged with Modbus Slave, page 68). The correspondence between the different elements in these tree structures and the standard Modbus terminology is shown on the right:



7. Using ABC-LUFP Configurator



N.B. In all cases, the “Query / Slave Address” and “Response / Slave Address” elements are automatically updated by AbcConf according to the node in which the command is located. Their values are not user-modifiable. Likewise, the “Query / Function” and “Response / Function” fields depend on the nature of the Modbus command and are not user-modifiable.

The operations to be carried out are much the same as those used for changing the default commands. For the “Read Holding Registers” command, see Section 7.9.1 Replacing a Periodic Input, page 68, and Section 7.9.3 Increasing the Number of Periodic Inputs, page 70. For the “Preset Multiple Registers” command, see Section 7.9.2 Replacing a Periodic Output, page 69, and Section 7.9.4 Increasing the Number of Periodic Outputs, page 74.

ENGLISH

7.11.2. With a Generic Modbus Slave

Unlike in the previous section, here we will be looking at adding and setting up a Modbus command which is different from those configured by default with the LUFP1 gateway. We will use this opportunity to take an in-depth look at the fields with which you can set up communications for such commands.

See Chapter 12 Appendix E: Modbus Commands, page 120, for the list and description of the Modbus functions supported by the LUFP1 gateway. Only a very few Modbus commands are authorised, in order to limit the complexity of configuring Modbus exchanges on the gateway. Moreover, AbcConf cannot be used to configure new Modbus commands (created from scratch) for LUFP1 gateways.

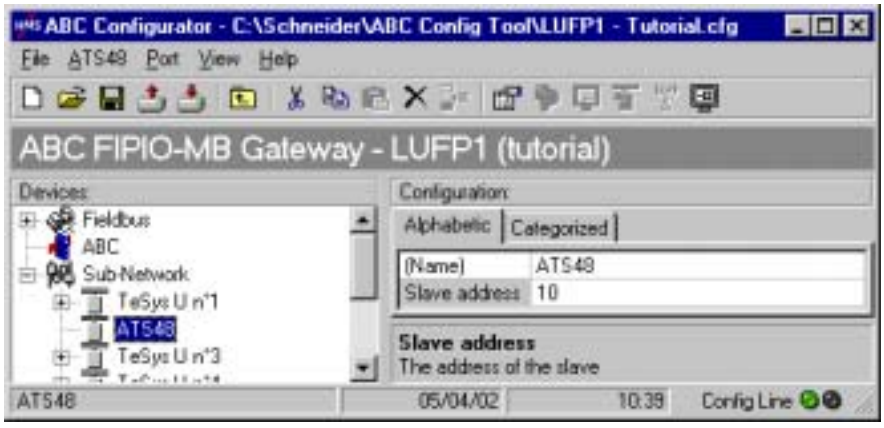
E.g. To illustrate the various operations to be carried out and the explanations given, we will be taking the example of a Altistart starter, the ATS48, and a Modbus command recognised both by the gateway and the ATS48. This is the “Preset Single Register” command, whose function code is 6 and which allows you to write the value of a single output word. This function will be used to periodically write the value of the ATS48’s CMD command register, located at address W400 (address 400 = 16#0190).

7. Using ABC-LUFP Configurator

As the standard gateway configuration already has 8 Modbus slaves, you will need to delete one of them, such as the “TeSys U n°2” node, for example, and to add a new node in its place (see Section 7.7 Deleting a Modbus Slave, page 66, and Section 7.8 Adding a Modbus Slave, page 67).

Here, we have just created “New Node”, in “ATS48” and assigned it Modbus address 10.

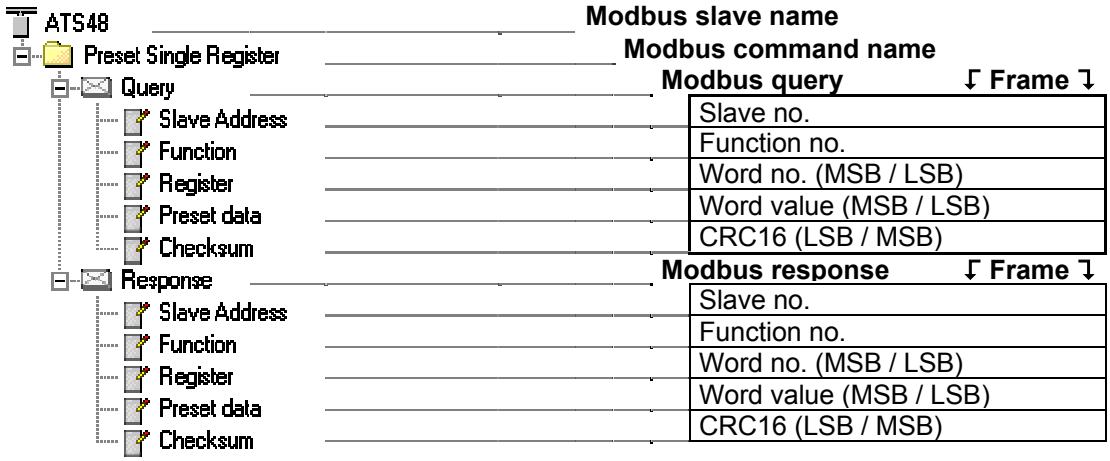
We then proceed to add the “Preset Single Register” command by choosing “Add Command” from the “ATS48” menu.



In the window which appears (opposite), select the command “0x06 Preset Single Register” and choose “Select” in the “File” menu.

Back in the main AbcConf window, the “Preset Single Register” command now appears in the list of Modbus commands for the “ATS48” node.

Open up the full tree structure for this command, as shown below. The correspondence between the different elements in this tree structure and the standard Modbus terminology is shown on the right:



These elements can be configured using AbcConf. There is a description of them in the following sections. We will then return to the example of the ATS48 to illustrate how to use these elements.

ENGLISH

7.11.2.1. Managing Degraded Modes

Due to the number of hardware elements and software tools used, the following table shows a summary of the various degraded modes available to a FIPIO application. In this instance, the application is running on a LUFP1 gateway and a Premium PLC with a FIPIO master coupler on its processor card.

Desired behaviour		Event			
		Premium PLC: CPU stop/failure	**Disconnection of the upstream FIPIO network (1)**	**LUFP1 gateway failure**	**Disconnection of the downstream Modbus RTU network (1) (3)**
Outputs	**Reset**	Yes	"Offline options for fieldbus" = "Clear"	Depends on the configuration of the Modbus slaves (2)	
	Hold	——	"Offline options for fieldbus" = "Freeze"		
	No refresh		"Offline options for fieldbus" = "NoScanning"	——	
Inputs	**Reset**	——	——		"Offline options for sub-network" = "Clear"
	Hold		Yes (4)		"Offline options for sub-network" = "Freeze"

(1) The "Offline options for fieldbus" and "Offline options for sub-network" are described in the next section.

(2) The desired behaviour with regard to outputs should be directly configured on each of the Modbus slaves. In the case of drives marketed by *Schneider Electric*, for instance, the outputs are reset by setting the NTO bit to 0 (command with communication control), and held by setting NTO to 1 (command without communication control).

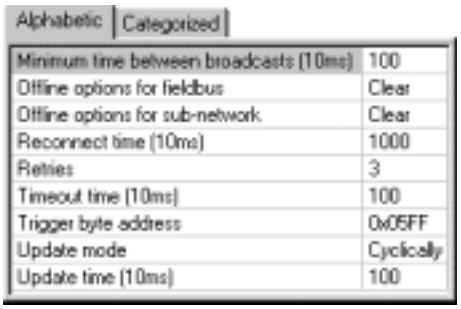
(3) As regards inputs, we strongly recommend that you use the List of Active Slaves (LAS) Service (see Section 5.2, page 42) to detect the underlying event.

(4) Use the gateway's FIPIO Diagnostic Objects (Section 5.4, page 52) to detect the underlying event.

ENGLISH

7.11.2.2. Configuring the Query

Select the "Query" element from the Modbus command. The various elements in the configuration of the query for this command are shown opposite. The values displayed correspond to the default values for any new command.



These elements allow you to configure how the whole command is managed, including how degraded modes are managed (number of retransmissions, for example).

Each of these elements is described, in order, in the table below. When a unit is assigned to an element, it is shown in brackets after the name of that element.

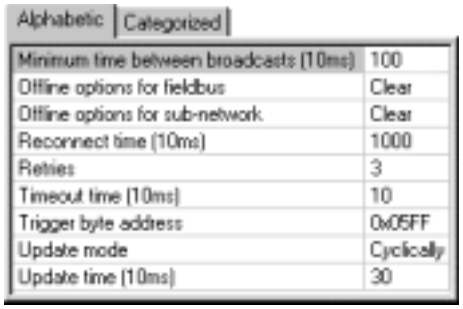
Configuration element	Description
Minimum time between broadcasts (10ms)	This configuration element is not used with the LUFP1 gateway: it is only activated if the command that contains it is a broadcast command, which is not possible with the LUFP1 gateway. **N.B.** The only way to run broadcast commands is by using the indexed periodic variables (PKW) service described in Section 5.3 Indexed Periodic Variables (PKW) Service, page 43, and only when the service is used in broadcast mode (DN = 16#FF).
Offline options for fieldbus	This element affects the data ***sent to the Modbus slave***, only for the command to which the element belongs, whenever the gateway is disconnected from the FIPIO network. It takes one of the following three values: - Clear ..... All data sent to the Modbus slave using this command is now set to 16#0000 (resetting the output data in the gateway's memory). - Freeze..... All data sent to the Modbus slave using this command retains its current value (freezing the output data in the gateway's memory). - NoScanning .. The command is no longer transmitted to the Modbus slave by the gateway.
Offline options for sub-network	This element affects the data ***sent to the FIPIO master PLC*** whenever the query to which this element belongs has not received a response from the Modbus slave (no response). It takes one of the following two values: - Clear ..... The data sent to the FIPIO master PLC is now set to 16#0000 (resetting the input data in the gateway's memory). - Freeze..... The data sent to the FIPIO master PLC retains its current value (freezing the input data in the gateway's memory). **N.B.** Exception responses do not trigger the use of these "Offline options" !
Reconnect time (10ms)	If there is no response from the Modbus slave to a query, or following the receipt of an incorrect response, the gateway uses the "Retries" and "Timeout time (10ms)" elements to carry out retransmissions. If the Modbus slave has still not responded correctly after these retransmissions, the gateway stops sending it the corresponding query for a period of time which can be adjusted using "Reconnect time (10ms)". When this period is over, the gateway attempts to restore communication with the Modbus slave.
Retries	This element indicates the number of retransmissions carried out by the gateway if there is no response from the Modbus slave to a query, or if the response is incorrect. This retransmission process ceases as soon as the gateway gets a correct response within a given time. If none of the retransmissions elicits a correct response, the Modbus slave is deemed to be off-line, but only as regards the command in question. The gateway then uses the "Offline options for sub-network" and "Reconnect time (10ms)" elements, and LED ❺ MODBUS becomes red. It will only revert to green when the Modbus command receives a correct response, following reconnection (see the element "Reconnect time (10ms)"). If the number of retransmissions is set to 0, this process will not be executed.
Timeout time (10ms)	This element represents the time that the Modbus slave will wait for a response. If a response has not reached the gateway within the given time, as defined by this element, the gateway tries retransmitting. This process continues until it reaches the last retransmission allowed (see "Retries"), then the gateway declares the Modbus slave off-line, but only for the command to which the element "timeout time (10ms)" belongs.

Configuration element	Description
Trigger byte address	This element is only used by the gateway if “Update mode” is set to “Change of state on trigger”. In this case, it specifies the address, in the gateway’s output memory (16#0200 to 16#0233), of an 8-bit counter managed by the FIPIO master. The FIPIO master updates this counter in the same way as the other gateway outputs (see Section 5.1.3 Configuration under AbcConf with Profile FED C32, page 41). When the value located at this address is changed by the FIPIO master, the query configured with a “Change of state on trigger” for that address is transmitted to the Modbus slave by the gateway. Unlike the “On data change” and “Change of state or Cyclically” modes, allows you to send a command on a specific order from the FIPIO master if, for example, the latter is unable to update all the data for any given query at the same time. **N.B.** The “trigger byte” does not have to be an output updated by the FIPIO master. It is quite possible for it to be an input between 16#0000 and 16#0033, in which case the exchanges of the command currently being configured will be conditioned by the Modbus slave that updates the byte.
Update mode	This element is used to specify the transmission mode for the query on the Modbus network. It takes one of the following four values: - Cyclically ..... Default communication mode. The query is transmitted **periodically** over the Modbus network (see “Update time”). **FED C32 P** This is the communication mode used systematically for all Modbus commands generated by the periodic control words configured by the user under PL7 PRO. - On data change ..... The gateway transmits the query over the Modbus network when at least one data item in the query is modified by the FIPIO master. This is therefore an **aperiodic** communication mode that works in a similar way to the indexed periodic variables (PKW) service described in Section 5.3 Indexed Periodic Variables (PKW) Service, page 43. All the data from a single query must therefore be updated at the same time by the FIPIO master. If you are uncertain as to whether your software is able to update **all the output data** from a query at the same time, we recommend using “Change of State on trigger” mode rather than “On data change”. - Single Shot ..... This transmission mode only allows a single Modbus exchange for the whole of the time that the gateway is operating. This exchange takes place just after the initialisation of the gateway. - Change of state on trigger ..... With this **aperiodic** communication mode, the Modbus query is sent every time that the FIPIO master changes the value of an 8-bit counter designated by the “Trigger byte address” element. See the description of this element for further information about how to use this communication mode. - Change of state or Cyclically .... In this communication mode—a combination of the “Cyclically” and “On data change” modes—the query is transmitted periodically over the Modbus network (see “Update time”), and also whenever one of the outputs in the query is modified by the FIPIO master. **FED C32 P** This is the communication mode used systematically for all Modbus commands generated by the periodic command words configured by the user under PL7 PRO.
Update time (10ms)	This element is only used by the gateway if “Update mode” is set to “Cyclically” or “Change of state or Cyclically”. In this case, it specifies the query’s transmission period on the Modbus network.

ENGLISH

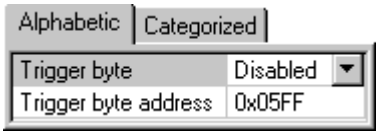
E.g. With the ATS48, we will be using the configuration shown opposite. The most notable points of this configuration are:

- The data are reset when either network is disconnected.
- 3 retransmissions with a 100 ms timeout.
- Periodic communications with a cycle time of 300 ms.



7.11.2.3. Configuring the Response

Select the "Response" element from the Modbus command. The various configuration elements of the response for this command are shown opposite. The values displayed are the default values for any new command.



These elements allow you to configure just one command management feature, described below. The elements are described, in order, in the following table:

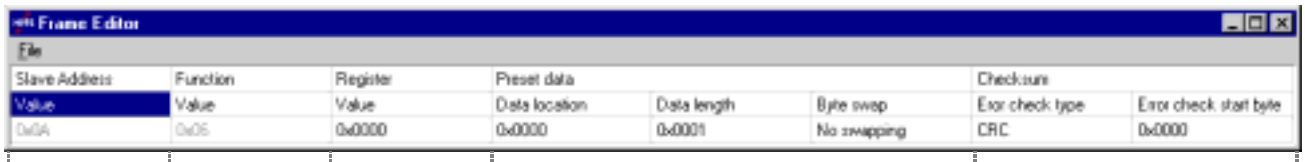
Configuration element	Description
Trigger byte	This element is used by the gateway to activate the unitary incrementation of an 8-bit counter in order to notify the FIPIO master of the receipt of a new response to the associated Modbus command. It takes one of the following two values: - Disabled ..... Default configuration. The gateway does not increment any counter on receipt of the Modbus response. - Enabled ..... Each time that the gateway receives a new response to the associated Modbus command, it increments the value of an 8-bit counter designated by the "Trigger byte address" element (see below). This is useful when the command query is configured to use one of the **aperiodic** communication modes (see "Update mode"), as the counter mechanism allows the FIPIO master to ignore response input data except when the counter is incremented.
Trigger byte address	This element is only used by the gateway if the element "Trigger byte" is set to "Enabled". In this case, it specifies the address, in the gateway's input memory (16#0000 to 16#0033), of an 8-bit counter managed by the gateway. The FIPIO master reads the value of this counter in the same way as the other gateway inputs (see Section 5.1.3 Configuration under AbcConf with Profile FED C32, page 41). When the gateway receives a response to the associated Modbus command, it increments the value of this counter by one unit (value = value+1). This mode keeps the FIPIO master informed when a new response is available. This can be useful, for example, where there is a chance that the data from two consecutive responses could be identical.

E.g.: With the ATS48, we do not want the response to be event driven, so we will retain the default configuration.

7. Using ABC-LUFP Configurator

7.11.2.4. Configuring the Content of the Query Frame

The window below is obtained using "Edit Frame" from the "Query" menu. Unlike the tree structure in the main AbcConf window, this display has the advantage of showing all of the frame's fields along with their values. The values displayed below are the values assigned by default to the Modbus command query we have created. The correspondence with the content of the corresponding Modbus frame is shown underneath.



Slave n°	Function n°	Word n° (MSB / LSB)	Word value (MSB / LSB)	CRC16 (LSB / MSB)

Edit the values which are not greyed out, one after the other. There is a description of them below.

The nature of a frame's fields depends on the Modbus command to which it corresponds. However, a certain number of these fields are common to all frames, whereas others are common to a number of them. The description of those shown above is given on the next page, as a part of the example described at the beginning of Section 7.11.2.

Field in the frame	Size in the frame	Description
Slave Address	1 byte	This field is not user-modifiable and its value is greyed out accordingly. AbcConf updates the value of this field automatically using the address of the Modbus slave corresponding to the current node. **N.B.** This field is common to all Modbus command queries. *Example:* The value of this field is set to the address of the Modbus slave corresponding to the "ATS48" node, namely 16#0A.
Function	1 byte	This field is not user-modifiable and its value is greyed out accordingly. AbcConf updates the value of this field automatically using the function code of the corresponding Modbus command. **N.B.** This field is common to all Modbus command queries. *E.g.* The value of this field is set to the code for the "Preset Single Register" command (write the value of an output word), namely 16#06.
Register	2 bytes	Address of an output word, or of a register, in the Modbus slave's memory. This field designates the memory object to which the command relates. **N.B.** This field is common to all Modbus command queries which seek to access one or more locations in the memory of a Modbus slave. When accessing several memory locations, the "Register" field designates the address of the first word affected by the command. *E.g.* The value of this field should be changed by entering the address of the CMD command register, namely 400 (16#0190). This value will be automatically converted to hexadecimal if the user enters it in decimal.

ENGLISH

Field in the frame	Size in the frame	Description
Preset Data	2 bytes, or more for a data block	Data Location: Address, in the gateway's output data memory (16#0200 to 16#0233), of the datum to be transmitted in the "Preset Data" field of the query frame. **N.B.** The field "Data location" is used for each frame that conveys data between the Modbus slaves and the FIPIO master. In this instance, it designates the starting address of the data block to be transmitted. **N.B.** As far as possible, ensure that the data is located at even-numbered addresses in order to align the Modbus data (in 16-bit format) with the %QW\p.2.c\0.0.•• outputs of the FIPIO master. *E.g.* The value to be assigned to the ATS48's CMD register should be placed in the gateway's output data memory area. We will be using the first free location starting at an even address, i.e. 16#0210 in the case of the standard configuration.
		Data length: Length of the output data block, in the gateway's memory, of which the values are to be transmitted in the "Preset Data" field of the query frame. It is expressed in number of bytes. **N.B.** The "Data length" field is always used together with the "Data location" field, described above. *E.g.* Since the "Preset Single Register" command is used to write the value of a single (16-bit) register, the value of the "Data length" field must be set to 2.  See the documentation for each Modbus slave to find out the maximum amount of 8-bit data which can be placed in "Data" type fields in the queries and responses for this slave. With the ATS48, for instance, the maximum is 30 16-bit words.
		Byte swap: Specifies whether or not the output data bytes to be transmitted to the Modbus slave must be swapped before being placed in the Modbus frame. The three possible values are as follows: - No swapping .....Default configuration. The data is sent in the same order as it appears in the gateway's memory. **This is the case which must be used by default**, because for a 16-bit datum, the most significant byte (MSB) is placed first in the Modbus frame, and is always written MSB-first into the gateway's memory by a FIPIO master. - Swap 2 bytes.....The bytes to be transmitted are swapped two by two. - Swap 4 bytes.....The bytes to be transmitted are swapped four by four. This is rarely used, as it only relates to 32-bit data. It works along the same lines as "Swap 2 bytes". *Example:* We will be using the "No swapping" value, because the two bytes of the value to be written into the ATS48's CMD register, as transmitted by the FIPIO master, are placed in the gateway memory MSB-first.

Field in the frame	Size in the frame	Description
Checksum	2 bytes	Error check type: Error checking method used for the frame. - CRC.....Default method. This is the method adopted for the Modbus RTU protocol. - LRC .....This method relates to the Modbus ASCII protocol, and so should not be used in this case. - XOR.....A simple "Exclusive OR" applied to the frame's bytes. *E.g.* The LUFP1 gateway is specifically designed for the Modbus protocol RTU mode. **The default value, "CRC", should not be changed.**
		Error check start byte: Indicates the number of the byte, in the frame, from which the "checksum" calculation should begin. The first byte in each frame carries the number 0. *E.g.* The calculation of a frame's checksum should always begin with the first byte. **The value of this field should therefore remain set to zero.**

7.11.2.5. Configuring the Content of the Response Frame

The window shown below is obtained using "Edit Frame" from the "Response" menu. The values displayed are those assigned by default to the Modbus command response we have created. The correspondence with the content of the corresponding Modbus frame is shown underneath.

Slave n°	Function n°	Word n° (MSB / LSB)	Word value (MSB / LSB)	CRC16 (LSB / MSB)

Edit the values which are not greyed out, one after the other.

There is a description of them below, but also see the previous section, as the nature of the content of response frames is very similar to that of the fields in Modbus query frames.

If the value of one of the fields in the response from a Modbus slave is different from that configured via AbcConf, the response will be rejected by the gateway. It will then proceed to retransmit the query, provided that at least one retransmission has been configured for the command in question (see 7.11.2.2 Configuring the Query, page 83). Of course, this remark does not concern the actual data, i.e. the Modbus frame fields configured using the "Data location," "Data length," and "Byte swap" elements.

Field in the frame	Size in the frame	Description
Slave Address	1 byte	Identical to the query's "Slave Address" field.
Function	1 byte	Identical to the query's "Function" field.
Register	2 bytes	Identical to the query's "Register" field, since the Modbus response of any "Preset Single Register" command is an echo to the corresponding query. Here you should also enter the address of the memory object to which the command relates. *E.g.* Enter the value 400, converted to 16#0190 by AbcConf.

ENGLISH

Field in the frame	Size in the frame	Description
Preset Data	2 bytes, or more for a data block	Data Location: Address, in the gateway's input data memory (16#0000 to 16#0033), of the datum received in the "Preset Data" field of the response frame. **N.B.** As far as possible, ensure that the data is located at even-numbered addresses in order to align the Modbus data (in 16-bit format) with the %IW\p.2.c\0.0.•• inputs of the FIPIO master. *E.g.* The value sent back as an echo to the command must be placed in the gateway's input data memory area. We shall use the first two free bytes following the standard configuration input data, i.e. 16#0020-16#0021.
		Data length: Length of the block of input data received in the "Preset Data" field of the response frame. It is expressed in number of bytes. *E.g.* The value of the "Data length" field must be set to 2.
		Byte swap: Identical to the query's "Byte swap" field. *E.g.* We will also be using the "No swapping" value, for the same reasons as with the query.
Checksum	2 bytes	Error check type: Identical to the query's "Error check type" field.
		Error check start byte: Identical to the query's "Error check start byte" field.
		However, these two fields are not user-modifiable and their value are greyed out accordingly. AbcConf updates the values of these fields automatically using the values in the query's "Error check type" and "Error check start byte" fields.

ENGLISH

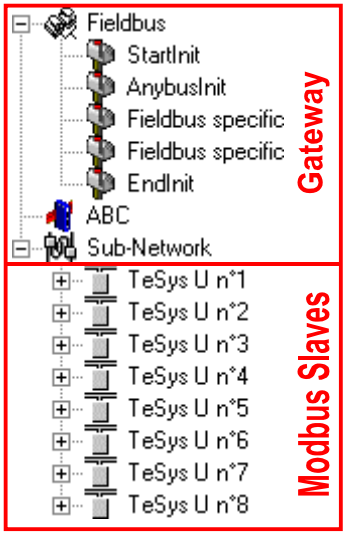
7.12. Configuring the General Characteristics of the Gateway

This operation relates to the gateway's general characteristics ("Fieldbus" to "Sub-Network" elements), whereas the previous sections described the configuration of the Modbus slaves (elements located under the "Sub-Network" element).

The "Fieldbus" element describes the upstream network, i.e. the FIPIO network in the case of the LUFP1 gateway.

The "ABC" and "Sub-Network" elements describe the downstream network—i.e. the Modbus network in the case of the LUFP1 gateway—and identify the version number of the gateway software.

The configuration of these three elements, along with the commands they give access to, are described in the next three sections.

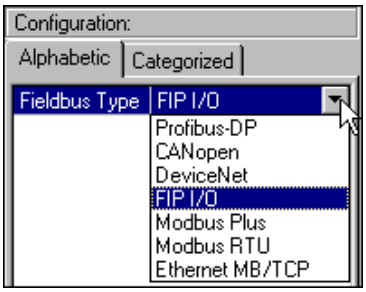


7. Using ABC-LUFP Configurator

7.12.1. "Fieldbus" Element

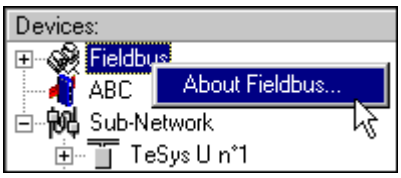
Below this element is a list of the mailboxes configured by default. These elements are not described here, as they are only designed for the internal management of the gateway. The mailboxes can neither be modified nor deleted. Their number and nature depend on the type of upstream network.

When the "Fieldbus" element is selected, you can select the type of upstream network. With the LUFP1 gateway, ***you must not modify the selection "FIP I / O"***.



If your PC is connected to the gateway using the PowerSuite cable and you are using AbcConf in "on-line" mode when AbcConf starts up, then the type of upstream network will be detected automatically.

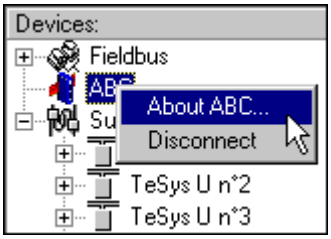
The only command accessible from the "Fieldbus" menu is "About Fieldbus…".

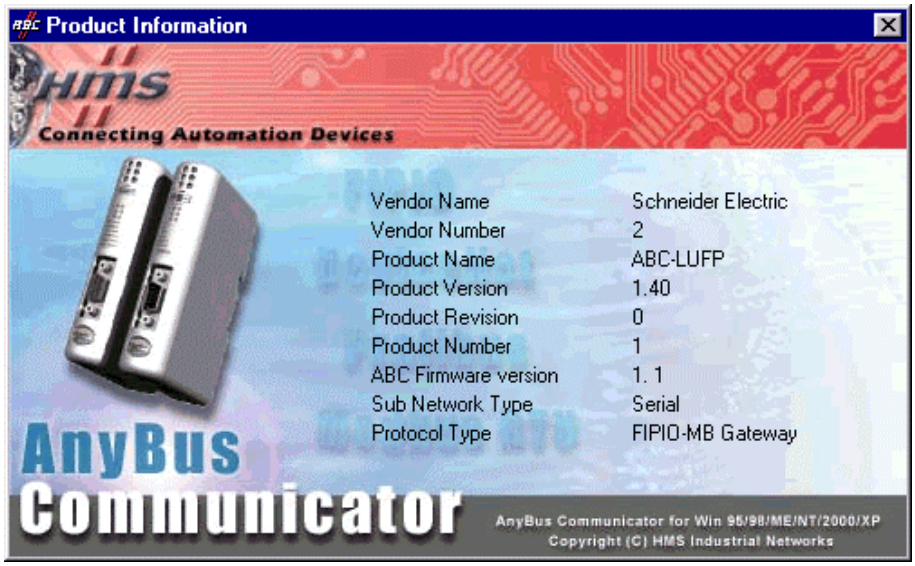


In "on-line" mode (see Section 7.12.2 "ABC" Element, page 91), the window shown opposite will be displayed. In "off-line" mode the word "Unknown" will replace "FIP I / O" to show that the type of upstream network cannot be identified.

7.12.2. "ABC" Element

The two commands accessible from the "ABC" menu are "About ABC-LUFP…" and "Disconnect" (or "Connect" if you are in "off-line" mode).





- Running "About ABC-LUFP…" allows AbcConf to retrieve and display all the version information about the software installed on the PC and on the gateway.

An example is shown opposite.

When you run "About ABC-LUFP…" in "off-line" mode, the last three fields are replaced by "Unknown" to show that the gateway software version cannot be identified.

ENGLISH

7. Using ABC-LUFP Configurator

N.B.: Only the version number of the software on the gateway's Modbus card is displayed. The gateway's FIPIO card software version is not accessible.

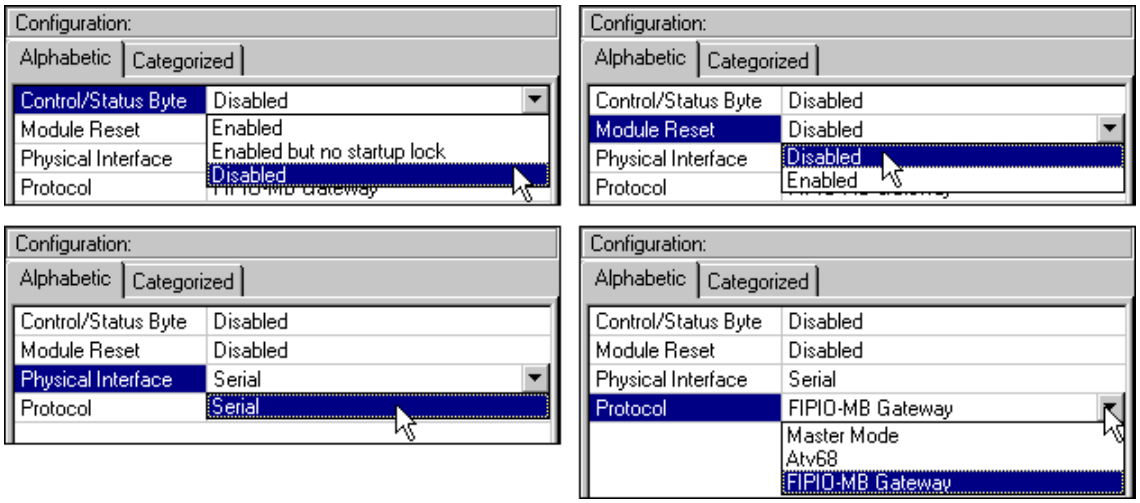
- The "Disconnect" command allows you to switch from "on-line" to "off-line" mode. It is only available in "on-line" mode. It is replaced by "Connect" once you are in "off-line" mode.

Apart from these two exclusive commands, the transition to "on-line" mode is requested by AbcConf when certain events occur (launch of AbcConf, use of "Upload" and "Download" commands, etc).

The AbcConf connection mode is displayed on the right of the status bar:

Config Line 	"On-line" mode (the LED on the left is green)
Config Line 	"Off-line" mode (the LED on the right is red)

Apart from the "Module Reset" option, ***the configuration of the LUFP1 gateway's "ABC" element should not be modified***. Of the four options shown below, the first, and the last two, should therefore retain the values: "Disabled", "Serial" and "FIPIO-MB Gateway" respectively.



These four options allow you to configure some of the gateway's system features:

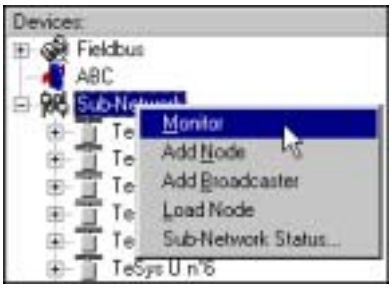
- Control / Status Byte: The three possibilities available for this option are not described in the LUFP1 Gateway User Manual, as the option is reserved for other devices in the same product family. ***This option must keep its existing value***, i.e. "Disabled".
- Module Reset: By default, this option prevents the gateway from reinitialising itself when there is an internal operation problem. Modifying this option is mainly intended for "laboratory" type use.
- Physical Interface: The only possibility offered by this option indicates that the physical interface of the downstream network of the gateway is a serial link.
- Protocol: This option should not be changed, because it indicates the type of protocol used on the network downstream of the gateway. With the LUFP1 gateway, "FIPIO-MB Gateway" ***must not be deselected***. The other possibilities available are reserved for other devices in the same product family.

Modifying the "Protocol" option will reset the entire configuration! It is therefore essential not to adjust it. If by any chance this should happen, start again with a clean basic configuration (see Section 7.6 Creating a New Configuration, page 66) or load a previously saved configuration.

ENGLISH

7.12.3. "Sub-Network" Element

The five commands accessible from the "Sub-Network" menu are:



- "Monitor": Allows you to view the mapping between the Modbus command data and the content of the gateway's memory. Examples of how to use this command are shown in Sections 7.9.3 (page 70) and 7.9.4 (page 74).

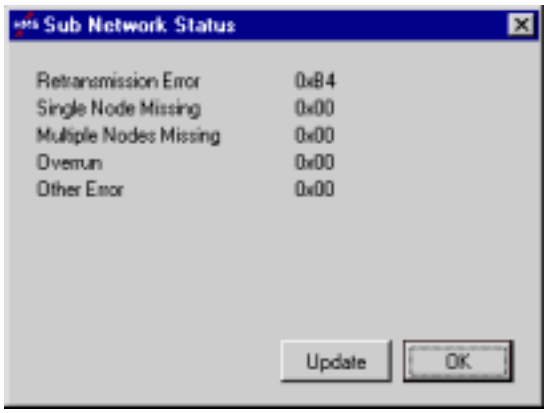
- "Add Node": Allows you to add a new node on the downstream Modbus network. Each node corresponds to a different Modbus slave. This command is not available if there are already 8 Modbus slaves, as is the case with the gateway's standard configuration.

- "Add Broadcaster": Allows you to add a broadcaster node (see Section 7.13 Adding a Broadcaster Node, page 95).

- "Load Node": Allows you to add a pre-configured node on the downstream Modbus network. The configuration for this node is contained in an XML file (see the section on "Importing/Exporting a Modbus Slave Configuration" in Section 7.8 Adding a Modbus Slave, page 67). This command is not available if there are already 8 Modbus slaves, as is the case with the gateway's standard configuration.

- "Sub-Network Status…": In "on-line" mode (see Section 7.12.2 "ABC" Element, page 91), this command opens a window displaying the values of the gateway's error counters. The "Update" button allows you to refresh the values of these counters.

  When you run this command in "off-line" mode, all of the values are replaced by the word "Unknown" to show that they cannot be read off the gateway. The "Update" button then becomes inaccessible.



When the "Sub-Network" element is selected, you have access to all of the options allowing you to configure the gateway's communication protocol format on the Modbus network. The various settings you can make are described below. All of the Modbus slaves present must support these settings and must be configured appropriately.

ENGLISH

7. Using ABC-LUFP Configurator

- Bitrate (bits/s): The gateway supports a limited number of communication speeds. Choose the one that suits your Modbus network.

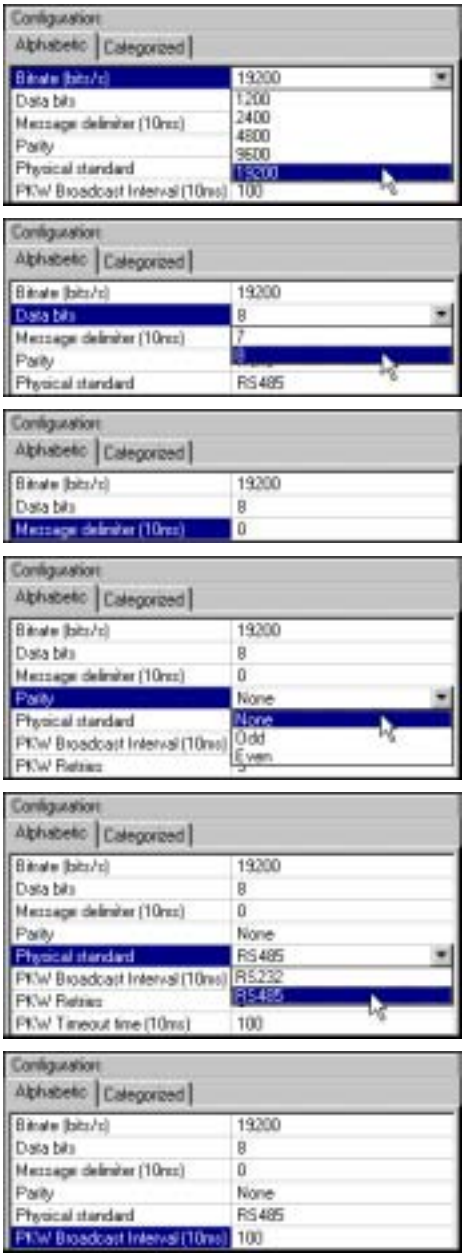
- Data bits: 8 bits (required).

- Message delimiter (10ms): Period of silence added to the normal silence time between the end of one message and the start of the next. The normal silence time corresponds to the time taken to transmit 3.5 characters.

- Parity: Choose the parity according to the format chosen for communications on your Modbus network.

- Physical standard: RS485 (required).

- PKW Broadcast Interval (10ms): When the PKW service is used in broadcast mode (DN = 255), this option defines the interval following the transmission of the broadcast command. The next Modbus message, of whatever type, will not be sent by the gateway until this period has elapsed. The interval must therefore be long enough to give the slowest Modbus slave the time to process the broadcast command.



- PKW Retries: Number of retransmissions attempted by the gateway when no response, or an erroneous response, is received from a queried Modbus slave using the PKW service. The retries stop as soon as the gateway receives a correct response.

- PKW Timeout time (10ms): Waiting time for a response from a Modbus slave using the PKW service. When the timeout elapses, the gateway starts a number of retries (see previous option).

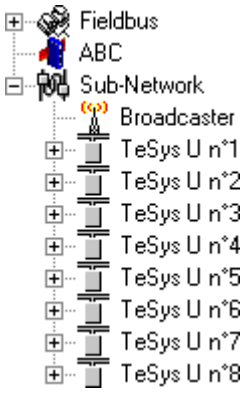
- Start bits: 1 bit (required).

- Stop bits: 1 or 2 bits.

7. Using ABC-LUFP Configurator

7.13. Adding a Broadcaster Node

N.B. In the specific case of the LUFP1 gateway, ***you cannot add Modbus commands to a broadcaster node***. If you want to broadcast a command across the Modbus network, we recommend that you use the PKW service in broadcast mode (DN = 255). See Section 5.3 Indexed Periodic Variables (PKW) Service, page 43, for more details about this service.



A broadcaster node does not correspond to any Modbus slave in particular: it applies to **all** Modbus slaves. All the commands configured for this node will be transmitted with the "Slave Address" field set to 16#00. This means that all of the slaves will execute the command, although none of them will respond to it.

To add a broadcaster node, select "Sub-Network", then choose "Add Broadcaster" from the "Sub-Network" menu (example opposite).

As the LUFP1 gateway does not support the adding of Modbus commands to a broadcaster node, we will not go into any further detail here about broadcaster nodes.

ENGLISH

8. Appendix A: Technical Characteristics

8.1. Environment

Dimensions (exc. connectors)	Height: 120 mm Width: 27 mm Depth: 75 mm
External appearance	Plastic case with device for fixing to a DIN rail.
Torque	PSU connector: between 5 and 7 lbs.-in.
Power supply	⎓ 24V insulated ±10% Maximum consumption: approx. 95 mA Maximum internal consumption for all of the gateway's electronic cards relating to the internal 5V PSU: 450 mA
Maximum relative humidity	95% without condensation or seepage, according to IEC 68-2-30
Ambient air temperature around the device, in a dry environment	According to IEC 68-2-1 Ab, IEC 68-2-2 Bb and IEC 68-2-14 Nb: • Storage: –25°C (±3) to +85°C (±2) • Operation: –05°C (±3) to +70°C (±2)
UL	E 214107 certificate "Open type" category The product should be installed in an electrical cabinet or equivalent location.
EC	Certified as complying with European standards, unless otherwise stated.
Electromagnetic compatibility (EMC): Transmission	Complies with industrial environment standard EN 50 081-2:1993. Tested for class A radiation under EN 55011:1990.
Electromagnetic compatibility (EMC): Immunity	Complies with industrial environment standards EN 50 082-2:1995 and EN 61 000-6-2:1999. Tested according to ENV 50 204:1995, EN 61000-4-2:1995, EN 61000-4-3:1996, EN 61000-4-4:1995, EN 61000-4-5:1995 and EN 61000-4-6:1996.

ENGLISH

8.2. Communication Characteristics

"Upstream" network	FIPIO
"Downstream" network	Modbus RTU
FIPIO characteristics	• Physical layer conforming to IEC IS 1158-2. • WorldFIP communication profile: Profile 2 (Device WorldFIP / FIPIO). • Nature of network: Device-oriented bus (DeviceBus). • Network topology: Multipoint linear topology (bus) with adapted and passive line terminations (see Section 2.6.2 Wiring Recommendations for the FIPIO Network, page 20). • Physical media (••• = 100, 200 or 500 m): - *Main cable:* 150 Ω shielded *single* twisted-pair copper cable: TSX FP CA••• - *Tap link cable:* 150 Ω shielded *double* twisted-pair copper cable: TSX FP CC••• • Connections: 9-pin SUB-D connectors preferred (see Section 2.6.2 Wiring Recommendations for the FIPIO Network, page 20). • Transmission rate: 1 Mbit/s. • Maximum length of an electrical segment: 1,000 m. • Maximum length of the FIPIO network: 15 km, in 15 segments of 1 km. • Maximum number of stations: 32 stations per segment, repeater not included; up to 128 stations for the entire FIPIO network (repeaters not included). Addresses 0 and 63 are reserved for the FIPIO master and the programming terminal respectively. • Mono-master network: Managed by a bus arbitrator (FIPIO master); the producer / consumer principle is used for inter-station exchanges. • Access methods: Periodic and aperiodic communications handled in bus "macro-cycle"; "time-critical" communication system.

8. Appendix A: Technical Characteristics

FIPIO characteristics (cont'd)	• FIPIO stations are configured according to standard exchange profiles (maximum of 32 words in input and in output for each profile). • Limited station configuration and adjustment possibilities using the FIPIO master. • Standard and customised station diagnostics. • Stations can be connected and disconnected without affecting communications between the other stations.
FIPIO specifics of LUFP1 gateway	• Communication profile (protocol): FIPIO (WorldFIP profile 2). • Communication profile (exchanges): - FED C32 P ... *Default profile*; configured by means of 30 configuration parameters and 30 adjustment parameters transmitted to the gateway by the FIPIO master on connection. - FED C32..... "Simplified" profile for the FIPIO master; configured using ABC-LUFP Configurator (AbcConf). Characteristics common to both profiles: 32 input words (%IW\p.2.c\0.0 to %IW\p.2.c\0.0.31), 32 output words (%QW\p.2.c\0.0 to %QW\p.2.c\0.0.31). • Standard connections: male 9-pin SUB-D connector. • Single transmission rate: 1 Mbit/s. • FIPIO station of "slave" type. • FIPIO address configured using 2 coding wheels (address between 1 and 99). In the case of Premium PLCs, addresses 0 and 63 are reserved. • Standard FIPIO diagnostics, but without "input validity". • Available inputs / outputs: 26 input words (%IW\p.2.c\0.0 to %IW\p.2.c\0.0.25) & 26 output words (%QW\p.2.c\0.0 to %QW\p.2.c\0.0.25) for profile FED C32; profile FED C32 P is limited to a sum total of 26 words. • List of active slaves (LAS) service occupies word %IW\p.2.c\0.0.27. • Indexed periodic variables (PKW) service occupies words %IW\p.2.c\0.0.28 to %IW\p.2.c\0.0.31 and %QW\p.2.c\0.0.28 to %QW\p.2.c\0.0.31.
Modbus RTU characteristics	• Physical media: RS485 serial link • Network topology: Multipoint linear topology with adapted line terminations (impedance of 120 Ω in parallel with a capacity of 1 nF) • Transmission rate: 1,200 to 57,600 kbits/s • Data bits: 8 • Subscriber addresses: 1 to 247. Address 0 reserved for broadcasting. Addresses 65, 126 and 127 reserved if *Schneider Electric* Speed Variation products are used on the same Modbus network. • Silence time: Equivalent to the transmission of 3.5 characters.
Modbus RTU specifics of LUFP1 gateway	• Maximum number of subscribers (excluding gateway): 8 Modbus slaves. • Maximum number of configurable commands: - FED C32 P ... Max. 26 periodic Modbus commands (read + write), each limited to a single word read or written, which in turn limits the total number of words read and written to 26. - FED C32..... 26 periodic and / or aperiodic Modbus commands, with no limit on the number of words read or written using any one command; all available input / outputs (26 input words and 26 output words) can therefore be used.

ENGLISH

8. Appendix A: Technical Characteristics

Modbus RTU specifics of LUFP1 gateway (cont'd)	• Characteristics configured under PL7 PRO (profile FED C32 P) or under AbcConf (profile FED C32): - Modbus read commands: Periodic mode mandatory (in FED C32 P) or user-configurable mode under AbcConf (in FED C32). - Modbus write commands: Periodic mode with transmission on data change (1) mandatory (in FED C32 P) or user-configurable mode under AbcConf (in FED C32). - Transmission rate: 1200, 2400, 4800, 9600 or 19200 bits/s. - Parity: None, even or odd. - Start bits: 1 bit. - Stop bits: 1 or 2 bits. - Silence time: The gateway's silence time can be incremented in 10 ms steps, ***but only under AbcConf***.

(1) Mode of communication (Update mode) for which the AbcConf equivalent is called "Change of state or Cyclically". The query for the command thus configured is transmitted periodically, but this periodicity is overridden if any of the data to be sent are modified. This command mode prioritises write commands, ensuring that they are sent as soon as possible.

LUFP1 gateway memory structure:

Inputs

- 52 bytes accessible to the FIPIO master in the form of input data (see Section 5.1.3 Configuration under AbcConf with Profile FED C32, page 41, for the mapping between the addresses of these bytes and the gateway's FIPIO objects).
- 460 input bytes inaccessible to the FIPIO master due to the use of profile FED C32 / FED C32 P, inputs reserved for the LAS and PKW services, and the presence of a reserved word.

Addresses	**Input data area**
16#0000 : 16#0033	Inputs accessible to the FIPIO master (52 bytes)
16#0034 : 16#01FF	Inputs inaccessible to the FIPIO master (460 bytes)

LUFP1 gateway memory structure:

Outputs

- 52 bytes accessible by the FIPIO master in the form of output data (see Section 5.1.3 Configuration under AbcConf with Profile FED C32, page 41, for the mapping between the addresses of these bytes and the gateway's FIPIO objects).
- 460 output bytes inaccessible to the FIPIO master due to the use of profile FED C32 / FED C32 P, outputs reserved for the LAS and PKW services, and the presence of two reserved words.

Addresses	**Output data area**
16#0200 : 16#0233	Outputs accessible to the FIPIO master (52 bytes)
16#0234 : 16#03FF	Outputs inaccessible to the FIPIO master (460 bytes)

8. Appendix A: Technical Characteristics

<table>
<tr>
<td>LUFP1 gateway memory structure:

General data</td>
<td>• 1024 bytes inaccessible to the FIPIO master.

<table>
<tr><th>Addresses</th><th>General data area</th></tr>
<tr><td>16#0400
16#051F</td><td>Input area reserved for the Mailboxes
(288 bytes)</td></tr>
<tr><td>16#0520
16#063F</td><td>Output area reserved for the Mailboxes
(288 bytes)</td></tr>
<tr><td>16#0640
16#07BF</td><td>Internal area reserved for the management
of the upstream network
(384 bytes; area not used by the LUFP1 gateway)</td></tr>
<tr><td>16#07C0
16#07FD</td><td>Internal area reserved for the control registers
(62 bytes / MSB first for 16-bit data)</td></tr>
<tr><td>16#07FE
16#07FF</td><td>Gateway status / FIPIO master command
(2 reserved bytes on the LUFP1 gateway)</td></tr>
</table>

You can use this data area to store data from a Modbus response that you don't want to pass on to the FIPIO master. In this case, always use 16#4000 as the starting address. If you use the same addresses more than once in this area, the corresponding locations will be displayed in red in the "General Area" pane of the "Sub-network Monitor" window (see page 72 for an example), but it will not affect the operation of the gateway.</td>
</tr>
<tr>
<td>Data transfer order (swapping)</td>
<td>• FIPIO network: LSB first and MSB last.
• LUFP1 gateway: MSB stored in the lowest memory address.
• Modbus RTU network: MSB first and LSB last.

→ The option which should be selected for Modbus data stored in the gateway's memory is "No swapping". This option relates to all "Data" and "Preset data" fields in the Modbus query and response frames.</td>
</tr>
</table>

ENGLISH

9. Appendix B: LUFP1 Gateway Settings

The settings described here correspond to the configuration and adjustment parameters for the LUFP1 gateway. **They only exist when the gateway is used according to standard profile FED C32 P.** The exchange modes for the corresponding FIPIO objects are described in Section 5.6 Parameters Specific to Profile FED C32 P, page 54. Once they have been downloaded to the gateway, the values of these parameters can be read and / or replaced using the gateway's PKW service (see Section 5.3.5 LUFP1 Gateway Internal Registers, page 49).

Some of the configuration and adjustment parameters cannot be modified after the initial setup of the gateway using the PLC.
See the tables in Chapters 9.1 Configuration Parameters and 9.2 Adjustment Parameters.

9.1. Configuration Parameters

The configuration parameters enable the gateway to configure the Modbus commands that it exchanges with the Modbus slaves. The configuration comprises:

① The number of periodic write commands (periodic command words) for each slave (0 to 9);

② The number of periodic read commands (periodic control words) for each slave (0 to 9);

③ The address of each periodic command or control word.

The total number of periodic words (command words + control words) is limited to 26.

Each **periodic command word** corresponds to a Modbus write command to a single register (function 16#06 "Preset Single Register"). This command is cyclical, with advance transmission of the write query if the new value for the register is modified. This exchange mode is equivalent to "Update mode = Change of state or cyclically" (see page 85).

Each **periodic control word** corresponds to a Modbus read command for several registers (function 16#03 "Read Holding Registers"), even though the command is used to read a single register. This command is strictly cyclical. This exchange mode is equivalent to "Update mode = Cyclically" (see page 85).

The configuration parameter values for the standard 8 TeSys U motor starter configuration are listed in Section 4.2.4 Configuration Parameter Values, Standard Config, page 29.

PL7 parameter	Access	Values	Description
%KW\p.2.c\0.0.0	RO	16#0000 to 16#FFFF	Number of periodic command words for slaves n°1 to n°4: Bits 0- 3: Number of periodic command words for slave n°1 = 0 to 15 Bits 4- 7: Number of periodic command words for slave n°2 = 0 to 15 Bits 8-11: Number of periodic command words for slave n°3 = 0 to 15 Bits 12-15: Number of periodic command words for slave n°4 = 0 to 15
%KW\p.2.c\0.0.1	RO	16#0000 to 16#FFFF	Number of periodic command words for slaves n°5 to n°8: Bits 0- 3: Number of periodic command words for slave n°5 = 0 to 15 Bits 4- 7: Number of periodic command words for slave n°6 = 0 to 15 Bits 8-11: Number of periodic command words for slave n°7 = 0 to 15 Bits 12-15: Number of periodic command words for slave n°8 = 0 to 15
%KW\p.2.c\0.0.2	RO	16#0000 to 16#FFFF	Number of periodic control words for slaves n°1 to n°4: Bits 0- 3: Number of periodic control words for slave n°1 = 0 to 15 Bits 4- 7: Number of periodic control words for slave n°2 = 0 to 15 Bits 8-11: Number of periodic control words for slave n°3 = 0 to 15 Bits 12-15: Number of periodic control words for slave n°4 = 0 to 15
%KW\p.2.c\0.0.3	RO	16#0000 to 16#FFFF	Number of periodic control words for slaves n°5 to n°8: Bits 0- 3: Number of periodic control words for slave n°5 = 0 to 15 Bits 4- 7: Number of periodic control words for slave n°6 = 0 to 15 Bits 8-11: Number of periodic control words for slave n°7 = 0 to 15 Bits 12-15: Number of periodic control words for slave n°8 = 0 to 15

9. Appendix B: LUFP1 Gateway Settings

Parameter PL7	Access	Values	Description
%KW\p.2.c\0.0.4	R/W	16#••••	Address of periodic word n° 1
%KW\p.2.c\0.0.5	R/W	16#••••	Address of periodic word n° 2
%KW\p.2.c\0.0.6	R/W	16#••••	Address of periodic word n° 3
%KW\p.2.c\0.0.7	R/W	16#••••	Address of periodic word n° 4
%KW\p.2.c\0.0.8	R/W	16#••••	Address of periodic word n° 5
%KW\p.2.c\0.0.9	R/W	16#••••	Address of periodic word n° 6
%KW\p.2.c\0.0.10	R/W	16#••••	Address of periodic word n° 7
%KW\p.2.c\0.0.11	R/W	16#••••	Address of periodic word n° 8
%KW\p.2.c\0.0.12	R/W	16#••••	Address of periodic word n° 9
%KW\p.2.c\0.0.13	R/W	16#••••	Address of periodic word n°10
%KW\p.2.c\0.0.14	R/W	16#••••	Address of periodic word n°11
%KW\p.2.c\0.0.15	R/W	16#••••	Address of periodic word n°12
%KW\p.2.c\0.0.16	R/W	16#••••	Address of periodic word n°13

Parameter PL7	Access	Values	Description
%KW\p.2.c\0.0.17	R/W	16#••••	Address of periodic word n°14
%KW\p.2.c\0.0.18	R/W	16#••••	Address of periodic word n°15
%KW\p.2.c\0.0.19	R/W	16#••••	Address of periodic word n°16
%KW\p.2.c\0.0.20	R/W	16#••••	Address of periodic word n°17
%KW\p.2.c\0.0.21	R/W	16#••••	Address of periodic word n°18
%KW\p.2.c\0.0.22	R/W	16#••••	Address of periodic word n°19
%KW\p.2.c\0.0.23	R/W	16#••••	Address of periodic word n°20
%KW\p.2.c\0.0.24	R/W	16#••••	Address of periodic word n°21
%KW\p.2.c\0.0.25	R/W	16#••••	Address of periodic word n°22
%KW\p.2.c\0.0.26	R/W	16#••••	Address of periodic word n°23
%KW\p.2.c\0.0.27	R/W	16#••••	Address of periodic word n°24
%KW\p.2.c\0.0.28	R/W	16#••••	Address of periodic word n°25
%KW\p.2.c\0.0.29	R/W	16#••••	Address of periodic word n°26

In the above table, the mapping between the addresses of periodic words n°1 to 26 and the command and control words of the various Modbus slaves is not given, because it depends *entirely* on the numbers of periodic command and control words for the Modbus slaves as a whole. These periodic words are organised as follows:

Address of periodic word n°01	Addresses of the periodic command words for slave n°1
:	:
:	Addresses of the periodic command words for slave n°8
:	Addresses of the periodic control words for slave n°1
:	:
:	Addresses of the periodic control words for slave n°8
:	16#0000 (1)
:	:
Address of periodic word n°26	16#0000 (1)

(1) If you do not use all of the 26 periodic words available, you must reset the addresses of the unused periodic words.

9.2. Adjustment Parameters

The adjustment parameters enable the gateway to configure the general features of its exchanges with the Modbus slaves.

The adjustment parameters values for the standard 8 TeSys U motor starter configuration are listed in Section 4.2.5 Adjustment Parameter Values, Standard Configuration, page 30.

9. Appendix B: LUFP1 Gateway Settings

PL7 parameter	Access	Bits	Description		Possible values or range of values
%MW\p.2.c\0.0.20	RO	0- 7	Modbus speed		16#00 = 1200 bits/s 16#03 = 9600 bits/s 16#01 = 2400 bits/s 16#04 = 19200 bits/s 16#02 = 4800 bits/s
		8	Modbus format	Data bits	**0 = 7 bits (Modbus ASCII) → PROHIBITED** **1 = 8 bits (Modbus RTU) → MANDATORY**
		9		Stop bits	0 = 1 stop bit 1 = 2 stop bits
		10-11		Parity	Bit 11 = 0 and bit 10 = 0: No parity Bit 11 = 0 and bit 10 = 1: Even parity Bit 11 = 1 and bit 10 = 0: Odd parity
		12-15	Not used		2#0000
%MW\p.2.c\0.0.21	RO	0- 7	Modbus Timeout: Max. waiting period for a response from a Modbus slave. **N.B. Allow for the slave with the longest response time!**		1 to 255 (unit: 100 ms) = 100 ms to 25.5 s
		8-15	Number of retransmissions of a frame if no response from a Modbus slave. The slave is declared absent if these retransmissions are completed without a correct response being received from it.		0 = No retransmission 1 to 15 = Number of consecutive retransmissions of the same Modbus frame by the gateway
%MW\p.2.c\0.0.22	RO	0- 7	Reconnection time for a Modbus slave declared absent (see Number of retransmissions, above). The gateway stops communicating with the slave for the set duration.		1 to 255 (unit: 1 s) = 1 s to 255 s
		8-15	Not used		16#00
%MW\p.2.c\0.0.23	RO	0- 7	Effect on Modbus exchanges of disconnection of the FIPIO network, or non-refreshment of FIPIO output data.		16#00 = “Freeze” = Modbus exchanges continue normally; command words keep their current values. 16#01 = “NoScanning” = Stop all Modbus exchanges. 16#02 = “Clear” = Modbus exchanges continue, but command word values are reset to zero.
		8-15	Effect on FIPIO exchanges of disconnection of the Modbus network, or non-refreshment of input data.		16#00 = “No action” = Inputs keep their current values. 16#02 = “Clear” = Input values are reset to zero.
%MW\p.2.c\0.0.24	RO	0-15	Not used		16#0000
%MW\p.2.c\0.0.25	R/W	0- 7	Periodicity of Modbus exchanges	Fast commands	0 = Null periodicity; the Modbus command is generated as frequently as possible 1 to 255 (unit: 10 ms) = 10 ms to 2.55 s
		8-15		Normal commands	
%MW\p.2.c\0.0.26	R/W	0- 7		Fast controls	
		8-15		Normal controls	
%MW\p.2.c\0.0.27	R/W	0 : 7	Cycle time of periodic Modbus commands	Periodic word n° 1 …… Periodic word n° 8	0 = Fast command 1 = Normal command
		8 : 15		Periodic word n° 9 …… Periodic word n°16	
%MW\p.2.c\0.0.28	R/W	0 : 7		Periodic word n°17 …… Periodic word n°24	

ENGLISH

PL7 parameter	Access	Bits	Description		Possible values or range of values
		8 9		Periodic word n°25 Periodic word n°26	
		10-15		Not used	2#000000

ENGLISH

9. Appendix B: LUFP1 Gateway Settings

PL7 parameter	Access	Bits	Description		Possible values or range of values
%MW\p.2.c\0.0.29	R/W	0 : 7	Cycle time of periodic Modbus controls	Periodic word n° 1 …… Periodic word n° 8	0 = Fast control 1 = Normal control
		8 : 15		Periodic word n° 9 …… Periodic word n°16	
%MW\p.2.c\0.0.30	R/W	0 : 7		Periodic word n°17 …… Periodic word n°24	
		8 9		Periodic word n°25 Periodic word n°26	
		10-15		Not used	2#000000
%MW\p.2.c\0.0.31	RO	0- 7	Addresses of Modbus slaves	Modbus slave n°1	0 = No Modbus slave at this location 1-247 = Address of Modbus slave at this location (no "empty locations" are allowed between 2 slaves)
		8-15		Modbus slave n°2	
%MW\p.2.c\0.0.32	RO	0- 7		Modbus slave n°3	
		8-15		Modbus slave n°4	
%MW\p.2.c\0.0.33	RO	0- 7		Modbus slave n°5	
		8-15		Modbus slave n°6	
%MW\p.2.c\0.0.34	RO	0- 7		Modbus slave n°7	
		8-15		Modbus slave n°8	
%MW\p.2.c\0.0.35 …… %MW\p.2.c\0.0.49	RO	0-15	Not used		2#000000

ENGLISH

Periodicity of Modbus exchanges and **Cycle time of periodic Modbus commands / controls:** The parameters %MW\p.2.c\0.0.25 to %MW\p.2.c\0.0.30 set the cycle time of the Modbus commands corresponding to the periodic command and control words defined by the configuration parameters:

- The "periodicity of Modbus exchanges" parameters (%MW\p.2.c\0.0.25 and %MW\p.2.c\0.0.26) set fast or slow periodicities for the periodic command and control words.
- Each of the bits in the "cycle time of periodic Modbus commands" (%MW\p.2.c\0.0.27 and %MW\p.2.c\0.0.28) determines either fast periodicity (bit at 0) or slow periodicity (bit at 1) for the commands, both defined by the parameter %MW\p.2.c\0.0.25.
- Each of the bits in the "cycle time of periodic Modbus controls" parameters (%MW\p.2.c\0.0.29 and %MW\p.2.c\0.0.30) determines either fast periodicity (bit at 0) or slow periodicity (bit at 1) for the controls, both defined by the parameter %MW\p.2.c\0.0.26.

Addresses of Modbus slaves: Remember that you *must* define the Modbus slaves starting at Modbus slave n°1 and then on in sequence, without leaving any "empty slots".

9. Appendix B: LUFP1 Gateway Settings

9.3. Management of Degraded Modes

The management of degraded modes on the LUFP1 gateway, described in Section 4.1.4 Managing Degraded Modes, page 26, now depends on a gateway adjustment parameter. The table below is the same as the one in that section, except that the new parameter has been included:

Desired behaviour		Event			
		Premium PLC: CPU stop or failure	**Disconnection of the upstream FIPIO network (1)**	**Failure of the LUFP1 gateway**	**Disconnection of the downstream Modbus RTU network**
Outputs	**Reset**	Yes	%MW\p.2.c\0.0.23: X0-X7 = 16#02	Depending on the configuration of the Modbus slaves (2)	
	Hold	——	%MW\p.2.c\0.0.23: X0-X7 = 16#00		
	No refresh		%MW\p.2.c\0.0.23: X0-X7 = 16#01	——	
Inputs	**Reset**	——	——		Yes (3)
	Hold		Yes (4)		——

(1) See Section 9.2 Adjustment Parameters page 101, for a description of the adjustment parameter used in this column.

(2) The desired behaviour with regard to the outputs should be directly configured on each of the Modbus slaves. In the case of drives marketed by Schneider Electric, for instance, the outputs are reset by setting the NTO bit to 0 (command with communication control), and held by setting NTO to 1 (command Without communication control).

(3) As regards inputs, we strongly recommend that you use the List of Active Slaves (LAS) Service (see Section 5.2, page 42) to detect the underlying event.

(4) Use the FIPIO Diagnostic Objects (Section 5.4, page 52) to detect the underlying event.

ENGLISH

9.4. Sample Gateway Settings

The example below illustrates a system architecture and Modbus exchanges that differ from those in the standard configuration described in the rest of this manual.

The example uses the configuration and adjustment parameters described in the previous sections.

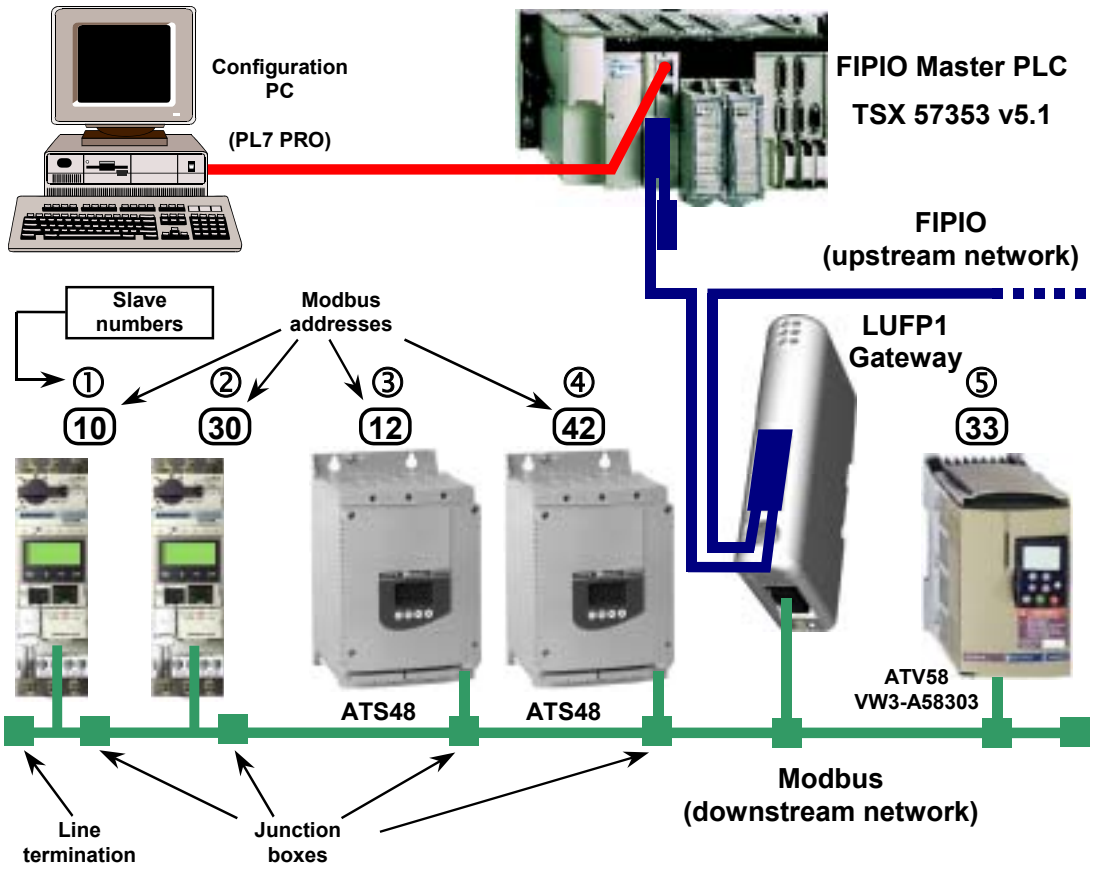
9.4.1. System Architecture

The Modbus slaves are grouped by category, rather than in order of address, in the list of Modbus slaves on the LUFP1 gateway. The slaves, as seen by the gateway, are as follows:

Slave	Product type	Product name	Modbus address
N°1	Motor starter	TeSys U	10
N°2		TeSys U	30
N°3	Soft start	Altistart 48	12
N°4		Altistart 48	42
N°5	Speed drive	Altivar 58 + Option VW3-A58303	33
N°6			0 (1)
N°7			0 (1)
N°8			0 (1)

(1) In the gateway settings, assigning a null address to a slave means that there is no slave at that location. Not to be confused with the broadcasting of Modbus commands!

9. Appendix B: LUFP1 Gateway Settings



ENGLISH

9.4.2. Modbus Exchanges

The table below lists all the registers read or written periodically by the gateway on the Modbus slaves. It will subsequently be used to configure the gateway's periodic command and control words.

Product	Exchange	Address	Register addressed	Periodicity
TeSys U (×2)	Command	704	Motor starter command	Fast
		700	Cassette command	Normal
	Control	455	Motor starter status	Fast
		452	Faults present	Normal
		461	Alarms present	Normal
ATS48 (×2)	Command	400	Command register	Fast
	Control	458	Status register	Fast
		4 072	Active power	Normal
		4 063	Torque	Normal
ATV58 (×1)	Command	400	DRIVECOM command register	Fast
		401	Online frequency setpoint (complement of 2)	Fast
		402	Internal command register (applications)	Normal
	Control	458	DRIVECOM status register	Fast
		451	Output frequency applied to motor (absolute value)	Normal
		452	Motor speed estimated by drive (absolute value)	Normal
		453	Motor current	Normal

9. Appendix B: LUFP1 Gateway Settings

Number of periodic command words = 2 (*2) + 1 (*2) + 3 (*1) = 4 + 2 + 3 = 9

Number of periodic control words = 3 (*2) + 3 (*2) + 4 (*1) = 6 + 6 + 4 = 16

Total number of periodic words = 9 + 16 = 25

Reserve: 1 periodic command or control word (maximum of 26 periodic words).

The periodic words are assigned firstly to the periodic command words, and then to the periodic control words. Each category starts with the first Modbus slave and ends with the last configured slave.

In the present example, this gives us the following configuration of periodic words; a grey background indicates a periodic word with a “fast” periodicity (otherwise the periodicity is “normal”):

Periodic word	Modbus slave	Register
1 Command n°1	N°1 TeSys U 10	704
2 Command n°2	N°1 TeSys U 10	700
3 Command n°3	N°2 TeSys U 30	
4 Command n°4	N°2 TeSys U 30	700
5 Command n°5	N°3 12	
6 Command n°6	ATS48 42	400
7 Command n°7	N°5 33	400
8 Command n°8	N°5 ATV58	401
9 Command n°9	N°5 ATV58 33	402
10 Control n° 1	TeSys U 10	455
11 Control n° 2	N°1 TeSys U 10	452
12 Control n° 3	N°1 TeSys U 10	461
13 Control n° 4	TeSys U 30	455

Periodic word	Modbus slave	Register
14 Control n° 5	N°2 TeSys U 30	452
15 Control n° 6	N°2 TeSys U 30	461
16 Control n° 7	N°3 ATS48 12	458
17 Control n° 8	N°3 ATS48 12	4 072
18 Control n° 9	N°3 ATS48 12	4 063
19 Control n°10	N°4 ATS48	458
20 Control n°11	N°4 ATS48 42	4 072
21 Control n°12	N°4 ATS48 42	4 063
22 Control n°13	ATV58	458
23 Control n°14	N°5 ATV58 33	451
24 Control n°15	N°5 ATV58 33	452
25 Control n°16	N°5 ATV58 33	453
26		

ENGLISH

General Modbus Configuration:

- Modbus speed ..... 19,200 bits/s
- Modbus format..... 8 data bits (Modbus RTU) — 1 stop bit — no parity
- Degraded modes ..... Timeout of 300 ms — 2 retransmissions — Reconnect after 30 s — Reset command values if FIPIO bus is disconnected
- Periodicities ..... Fast commands at 300 ms / normal at 900 ms
  Fast controls at 300 ms / normal at 1,800 ms

When you create a new configuration and you want to set up the gateway to handle the Modbus exchanges for this configuration, it is highly advisable to plot out all such exchanges in advance, as illustrated above.

N.B. The configuration of the Modbus slaves themselves is beyond the scope of this guide, and is therefore not described here. See the user manuals of the corresponding products.

9. Appendix B: LUFP1 Gateway Settings

9.4.3. Configuration Parameter Values

PL7 parameter	Value		Description
%KW\p.2.c\0.0.0	4 386	16#1122	Number of periodic command words for slaves n°1 to n°4: Bits 0- 3: Nbr of periodic command words for slave n°1 (TeSys U) = 2 Bits 4- 7: Nbr of periodic command words for slave n°2 (TeSys U) = 2 Bits 8-11: Nbr of periodic command words for slave n°3 (ATS48) = 1 Bits 12-15: Nbr of periodic command words for slave n°4 (ATS48) = 1
%KW\p.2.c\0.0.1	3	16#0003	Number of periodic command words for slaves n°5 to n°8: Bits 0- 3: Nbr of periodic command words for slave n°5 (ATV58) = 3 Bits 4- 7: Nbr of periodic command words for slave n°6 = 0 Bits 8-11: Nbr of periodic command words for slave n°7 = 0 Bits 12-15: Nbr of periodic command words for slave n°8 = 0
%KW\p.2.c\0.0.2	13 107	16#3333	Number of periodic control words for slaves n°1 to n°4: Bits 0- 3: Nbr of periodic control words for slave n°1 (TeSys U) = 3 Bits 4- 7: Nbr of periodic control words for slave n°2 (TeSys U) = 3 Bits 8-11: Nbr of periodic control words for slave n°3 (ATS48) = 3 Bits 12-15: Nbr of periodic control words for slave n°4 (ATS48) = 3
%KW\p.2.c\0.0.3	4	16#0004	Number of periodic control words for slaves n°5 to n°8: Bits 0- 3: Nbr of periodic control words for slave n°5 (ATV58) = 4 Bits 4- 7: Nbr of periodic control words for slave n°6 = 0 Bits 8-11: Nbr of periodic control words for slave n°7 = 0 Bits 12-15: Nbr of periodic control words for slave n°8 = 0
%KW\p.2.c\0.0.4	704	16#02C0	Address of periodic word n° 1: 1st prd. command word of slave n°1 (TeSys U)
%KW\p.2.c\0.0.5	700	16#02BC	Address of periodic word n°02: 2nd prd. command word of slave n°1 (TeSys U)
%KW\p.2.c\0.0.6	704	16#02C0	Address of periodic word n°03: 1st prd. command word of slave n°2 (TeSys U)
%KW\p.2.c\0.0.7	700	16#02BC	Address of periodic word n°04: 2nd prd. command word of slave n°2 (TeSys U)
%KW\p.2.c\0.0.8	400	16#0190	Address of periodic word n°05: 1st prd. command word of slave n°3 (ATS48)
%KW\p.2.c\0.0.9	400	16#0190	Address of periodic word n°06: 1st prd. command word of slave n°4 (ATS48)
%KW\p.2.c\0.0.10	400	16#0190	Address of periodic word n°07: 1st prd. command word of slave n°5 (ATV58)
%KW\p.2.c\0.0.11	401	16#0191	Address of periodic word n°08: 2nd prd. command word of slave n°5 (ATV58)
%KW\p.2.c\0.0.12	402	16#0192	Address of periodic word n°09: 3rd prd. command word of slave n°5 (ATV58)
%KW\p.2.c\0.0.13	455	16#01C7	Address of periodic word n°10: 1st prd. control word of slave n°1 (TeSys U)
%KW\p.2.c\0.0.14	452	16#01C4	Address of periodic word n°11: 2nd prd. control word of slave n°1 (TeSys U)
%KW\p.2.c\0.0.15	461	16#01CD	Address of periodic word n°12: 3rd prd. control word of slave n°1 (TeSys U)
%KW\p.2.c\0.0.16	455	16#01C7	Address of periodic word n°13: 1st prd. control word of slave n°2 (TeSys U)
%KW\p.2.c\0.0.17	452	16#01C4	Address of periodic word n°14: 2nd prd. control word of slave n°2 (TeSys U)
%KW\p.2.c\0.0.18	461	16#01CD	Address of periodic word n°15: 3rd prd. control word of slave n°2 (TeSys U)
%KW\p.2.c\0.0.19	458	16#01CA	Address of periodic word n°16: 1st prd. control word of slave n°3 (ATS48)
%KW\p.2.c\0.0.20	4 072	16#0FE8	Address of periodic word n°17: 2nd prd. control word of slave n°3 (ATS48)
%KW\p.2.c\0.0.21	4 063	16#0FDF	Address of periodic word n°18: 3rd prd. control word of slave n°3 (ATS48)
%KW\p.2.c\0.0.22	458	16#01CA	Address of periodic word n°19: 1st prd. control word of slave n°4 (ATS48)
%KW\p.2.c\0.0.23	4 072	16#0FE8	Address of periodic word n°20: 2nd prd. control word of slave n°4 (ATS48)
%KW\p.2.c\0.0.24	4 063	16#0FDF	Address of periodic word n°21: 3rd prd. control word of slave n°4 (ATS48)
%KW\p.2.c\0.0.25	458	16#01CA	Address of periodic word n°22: 1st prd. control word of slave n°5 (ATV58)
%KW\p.2.c\0.0.26	451	16#01C3	Address of periodic word n°23: 2nd prd. control word of slave n°5 (ATV58)
%KW\p.2.c\0.0.27	452	16#01C4	Address of periodic word n°24: 3rd prd. control word of slave n°5 (ATV58)
%KW\p.2.c\0.0.28	453	16#01C5	Address of periodic word n°25: 4th prd. control word of slave n°5 (ATV58)
%KW\p.2.c\0.0.29	0	16#0000	Address of periodic word n°26: Not used (reserved)

ENGLISH

9. Appendix B: LUFP1 Gateway Settings

9.4.4. Adjustment Parameter Values

Parameter	Value		Description
%MW\p.2.c\0.0.20	260	16#0104	Bits 0- 7: Modbus speed = 19,200 bits/s (LSB byte = 4) Bit 8: Modbus format = 8 data bits (bit at 1) Bit 9: Modbus format = 1 stop bit (bit at 0) Bits 10-11: Modbus format = No parity (bits at 0) Bits 12-15: Not used (bits at 0)
%MW\p.2.c\0.0.21	515	16#0203	Bits 0- 7: Modbus response timeout = 300 ms (3 * 100 ms) Bits 8-15: Number of frame retransmissions over the Modbus network = 2
%MW\p.2.c\0.0.22	30	16#001E	Bits 0- 7: Time taken by gateway to reconnect Modbus slave following a break in communication = 30 s (30 * 1 s) Bits 8-15: Not used (bits at 0)
%MW\p.2.c\0.0.23	514	16#0202	Bits 0- 7: Behaviour on losing FIPIO network = Reset periodic command word values transmitted to Modbus slaves (LSB byte = 2) Bits 8-15: Behaviour on losing Modbus network = Reset periodic control word values transmitted by Modbus slaves (LSB byte = 2)
%MW\p.2.c\0.0.24	0	16#0000	Not used (bits at 0)
%MW\p.2.c\0.0.25	23 070	16#5A1E	Bits 0- 7: Cycle time of fast periodic commands = 300 ms (30 * 10 ms) Bits 8-15: Cycle time of normal periodic commands = 900 ms (90 * 10 ms)
%MW\p.2.c\0.0.26	46 110	16#B41E	Bits 0- 7: Cycle time of fast periodic controls = 300 ms (30 * 10 ms) Bits 8-15: Cycle time of normal periodic controls = 1,800 ms (180 * 10 ms)
%MW\p.2.c\0.0.27	266	2#0000 0001 0000 1010	Cycle time settings for periodic commands n°1 to 16: Bit 0: Prd. command n°1 = fast (0) — Bit 5: Prd. command n°6 = fast (0) Bit 1: Prd. command n°2 = normal (1) — Bit 6: Prd. command n°7 = fast (0) Bit 2: Prd. command n°3 = fast (0) — Bit 7: Prd. command n°8 = fast (0) Bit 3: Prd. command n°4 = normal (1) — Bit 8: Prd. command n°9 = normal (1) Bit 4: Prd. command n°5 = fast (0) — Bits 9 to 15: Not used (bits at 0)
%MW\p.2.c\0.0.28	0	16#0000	Cycle time settings for periodic commands n°17 to 26: Not used
%MW\p.2.c\0.0.29	60 854	2#1110 1101 1011 0110	Cycle time settings for periodic controls n°1 to 16: Bit 0: Prd. control n° 1 = fast (0) — Bit 8: Prd. control n° 9 = normal (1) Bit 1: Prd. control n° 2 = normal (1) — Bit 9: Prd. control n°10 = fast (0) Bit 2: Prd. control n° 3 = normal (1) — Bit 10: Prd. control n°11 = normal (1) Bit 3: Prd. control n° 4 = fast (0) — Bit 11: Prd. control n°12 = normal (1) Bit 4: Prd. control n° 5 = normal (1) — Bit 12: Prd. control n°13 = fast (0) Bit 5: Prd. control n° 6 = normal (1) — Bit 13: Prd. control n°14 = normal (1) Bit 6: Prd. control n° 7 = fast (0) — Bit 14: Prd. control n°15 = normal (1) Bit 7: Prd. control n° 8 = normal (1) — Bit 15: Prd. control n°16 = normal (1)
%MW\p.2.c\0.0.30	0	16#0000	Cycle time settings for periodic controls n°17 to 26: Not used
%MW\p.2.c\0.0.31	7 690	16#1E0A	**Modbus addresses:** Slave n°1 (bits 0-7) = 10 / Slave n°2 (bits 8-15) = 30
%MW\p.2.c\0.0.32	10 764	16#2A0C	**Modbus addresses:** Slave n°3 (bits 0-7) = 12 / Slave n°4 (bits 8-15) = 42
%MW\p.2.c\0.0.33	33	16#0021	**Modbus addresses:** Slave n°5 (bits 0-7) = 33 / Slave n°6 (bits 8-15) = 0
%MW\p.2.c\0.0.34	0	16#0000	**Modbus addresses:** Slave n°7 (bits 0-7) = 0 / Slave n°8 (bits 8-15) = 0
%MW\p.2.c\0.0.35 …… %MW\p.2.c\0.0.49	0 …… 0	16#0000 …… 16#0000	Not used (bits at 0) …… Not used (bits at 0)

ENGLISH

9. Appendix B: LUFP1 Gateway Settings

9.4.5. FIPIO Objects Available for Programming

Only the Modbus slave control words (%IW\p.2.c\0.0 to %IW\p.2.c\0.0.26), the Modbus slave command words (%QW\p.2.c\0.0 to %QW\p.2.c\0.0.27) and the LUFP1 gateway LAS service (%IW\p.2.c\0.0.27) are discussed again here, as no other FIPIO objects are modified in the following example.

See Section 5 FIPIO Objects Available for Programming, page 38, for a description of all the gateway's FIPIO objects.

Inputs %IW\p.2.c\0.0 to %IW\p.2.c\0.0.26 correspond to the gateway's 26 periodic control words, plus one reserved word. As only control words n°1 to 16 are used, the only useful inputs are %IW\p.2.c\0.0 to %IW\p.2.c\0.0.15. Inputs %IW\p.2.c\0.0.16 to %IW\p.2.c\0.0.26 must therefore not be used:

LUFP1 input	Corresponding periodic word	Corresponding slave and exchange (read)
%IW\p.2.c\0.0 …… %IW\p.2.c\0.0.15	Periodic control word n° 1 …… Periodic control word n°16	Slave n°1 (TeSys U @10): Status of motor starter …… Slave n°5 (ATV58 @33): Current in motor
%IW\p.2.c\0.0.16 …… %IW\p.2.c\0.0.25	Periodic control word n°17 …… Periodic control word n°26	11 words not used (inputs at value 16#0000)
%IW\p.2.c\0.0.26	Reserved word	

Outputs %QW\p.2.c\0.0 to %QW\p.2.c\0.0.27 correspond to the gateway's 26 periodic command words, plus two reserved words. As only command words n°1 to 9 are used, the only useful outputs are %QW\p.2.c\0.0 to %QW\p.2.c\0.0.8. Outputs %QW\p.2.c\0.0.9 to %IW\p.2.c\0.0.27 must therefore not be used:

ENGLISH

LUFP1 output	Corresponding periodic word	Corresponding slave and exchange (write)
%IW\p.2.c\0.0 …… %IW\p.2.c\0.0.8	Periodic command word n° 1 …… Periodic command word n° 9	Slave n°1 (TeSys U @10): Motor starter command …… Slave n°5 (ATV58 @33): Internal command reg. (app)
%IW\p.2.c\0.0.9 …… %IW\p.2.c\0.0.25	Periodic command word n°10 …… Periodic command word n°26	19 words not used by the gateway
%IW\p.2.c\0.0.26 %IW\p.2.c\0.0.27	Reserved words	

N.B. Only the mapping for the first and last inputs (or outputs) is shown here, as the intervening periodic words were described in the tables in Section 9.4.2 Modbus Exchanges, page 106.

Only the first 5 bits of the input word assigned to the list of active slaves (LAS) service are actually used. A Modbus slave is present if the corresponding bit is set to 1.

See also Section 5.2 List of Active Slaves (LAS) Service, page 42, for more details on the list of active slaves service.

LAS service (%IW\p.2.c\0.0.27)

Bit	Modbus slave	Address
X0	TeSys U motor starter	10
X1	TeSys U motor starter	30
X2	Altistart 48 soft start	12
X3	Altistart 48 soft start	42
X4	Altivar 58 speed drive	33
X5-X15	Not used (bits at 0)	—

10. Appendix C: Standard Configuration

The configuration described below is the standard configuration used when implementing the LUFP1 gateway.

This chapter mainly gives the user information about the performances obtained on the downstream Modbus network. It allows users to decide whether, for example, they should change the timing of periodic exchanges with one or more of the TeSys U motor starters (see Chapter 7 Using ABC-LUFP Configurator, page 61).

10.1. Configuring Modbus Exchanges

The LUFP1 gateway carries out four types of exchanges with each of the 8 TeSys U motor starters. The first two exchanges are periodic and allow you to control and monitor the motor starter. The last two exchanges are aperiodic (triggered by a change in the values of the data to be transmitted to the motor starter) and allow you to read and change the value of any motor starter parameter.

Function	Modbus function	Number of bytes (1)	Exchange between the LUFP1 gateway and the TeSys U motor starter
16#03	Read Holding Register	11.5 + 10.5	Periodic reading (300 ms period) of the TeSys U motor starter's status register (address 455 = 16#01C7) only
16#10	Preset Multiple Registers	14.5 + 11.5	Periodic writing (300 ms period) of the TeSys U motor starter's status register (address 704 = 16#02C0) only
(16#03)	(Read Holding Register)	11.5 + 10.5	Aperiodic reading of the value of a single parameter, for a single TeSys U motor starter at a time (function and address defined by the user)
(16#06)	(Preset Single Register)	11.5 + 11.5	Aperiodic writing of the value of a single parameter, for a single TeSys U motor starter at a time (function and address and value defined by the user)

(1) Number of bytes in the Query + number of bytes in the Response, plus a silence time of 3.5 characters for each of these two frames (see description of the "Message delimiter (10ms)" parameter in Section 7.12.3 "Sub-Network" Element, page 93). Each byte will be transmitted in the form of a group of 10 bits (8 data bits, 1 start bit and 1 stop bit). These values allow you to calculate the approximate amount of traffic on the downstream Modbus network as follows:

Volume of periodic traffic (300 ms period) ..... *[(11.5 + 10.5) + (14.5 + 11.5)] * (8 + 1 + 1) = 480 bits*

For 1 TeSys U motor starter ..... *1 * 480 * (1 000 ÷ 300) = 1,600 bits/s*
For 8 TeSys U motor starters..... *8 * 480 * (1 000 ÷ 300) = 12,800 bits/s*

As a result, on a network operating at 9,600 bits/s, you will need to considerably increase the cycle time for all or some of the periodic Modbus commands. On the other hand, at a speed of 19,200 bits/s (the default speed), the available bandwidth is sufficient to support acceptable communications, even with the occasional degraded mode (retransmission of frames), and to allow the use of aperiodic parameter exchanges.

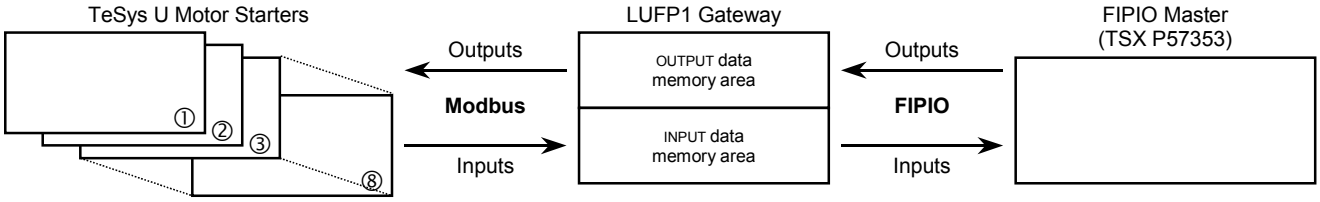
ENGLISH

10. Appendix C: Standard Configuration

10.2. Content of the Gateway's DPRAM Memory

The LUFP1 gateway's DPRAM memory contains all of the data exchanged between the gateway and the 8 TeSys U motor starters.

The flow of data between the TeSys U motor starters, the gateway and the FIPIO master is shown below, in order to highlight the role of the gateway's memory in these exchanges:



FED C32 P Configuring the gateway with this profile limits the total number of available words to 26, i.e. 52 bytes from among the 52 input bytes and the 52 output bytes.

FED C32 This profile makes available all of the gateway's 52 input bytes and 52 output bytes.

10.2.1. Input Data Memory Area

The gateway has 52 input bytes. Only the first 16 bytes are used.

Service	Address	Size	Description
Periodic communications — Control of TeSys U motor starters	16#0000	1 word	Value of status register for motor starter ①
	16#0002	1 word	Value of status register for motor starter ②
	16#0004	1 word	Value of status register for motor starter ③
	16#0006	1 word	Value of status register for motor starter ④
	16#0008	1 word	Value of status register for motor starter ⑤
	16#000A	1 word	Value of status register for motor starter ⑥
	16#000C	1 word	Value of status register for motor starter ⑦
	16#000E	1 word	Value of status register for motor starter ⑧
____	16#0010 … 16#0033	1 byte … 1 byte	Free input area (36 bytes)
_____	16#0034 16#0035	1 byte	Reserved memory locations (2 bytes, always at 16#0000)
LAS service (1)	16#0036	1 word	Reserved memory locations (2 bytes)
PKW service (RESPONSE) (1)	16#0038 … 16#003F	1 byte … 1 byte	Reserved memory locations (8 bytes)
____	16#0040 … 16#01FF	1 byte … 1 byte	Unusable input area (448 bytes)

(1) The data for these services are not actually stored in these gateway memory locations, but the mapping between the gateway memory and the corresponding FIPIO inputs / outputs (see Section 5.1.3 Configuration under AbcConf with Profile FED C32, page 41) means that these locations are not to be used. If they were, the mapping would overwrite the stored data with data from the services.

10. Appendix C: Standard Configuration

10.2.2. Output Data Memory Area

The gateway has 52 output bytes. Only the first 16 bytes are used.

Service	Address	Size	Description
Periodic communications — Command of TeSys U motor starters	16#0200	1 word	Value of command register for motor starter ①
	16#0202	1 word	Value of command register for motor starter ②
	16#0204	1 word	Value of command register for motor starter ③
	16#0206	1 word	Value of command register for motor starter ④
	16#0208	1 word	Value of command register for motor starter ⑤
	16#020A	1 word	Value of command register for motor starter ⑥
	16#020C	1 word	Value of command register for motor starter ⑦
	16#020E	1 word	Value of command register for motor starter ⑧
____	16#0210 … 16#0233	1 byte … 1 byte	Free output area (36 bytes)
_____	16#0234 … 16#0237	1 byte … 1 byte	Reserved memory locations (4 bytes not used by gateway)
PKW service (COMMAND) (1)	16#0238 … 16#023F	… 1 byte	Reserved memory locations (8 bytes)
____	16#0240 … 16#03FF	1 byte … 1 byte	Unusable output area (448 bytes)

(1) See note on “PKW service (RESPONSE)” in the previous table.

ENGLISH

10.2.3. Total Number of Modbus Queries and Responses

The total number of Modbus queries and responses is 32 (2 periodic queries and 2 periodic responses for each of the 8 TeSys U motor starters). Since the total number of Modbus queries and responses one can configure for a single gateway is limited to 52 (whichever profile is used: FED C32 or FED C32 P), there are 20 spare Modbus queries and responses (i.e. the equivalent of 10 Modbus commands).

This reserve is therefore enough for the addition of any single Modbus command to each of the TeSys U motor starters, as this requires 16 Modbus queries and responses (1 query and 1 response for each of the 8 motor starters).

11. Appendix D: Sample Use under PL7 PRO

A practical example can be found on CD LU9CD1.

- The first file, "LUFP1_FEDC32_Example.cfg", is an AbcConf file; its content corresponds to the system architecture described in Section 4.1.1 System Architecture, page 25. The gateway must therefore be configured using profile **FED C32** under PL7 PRO (see Chapter 6 Advanced Implementation of the Gateway, page 56). This AbcConf file corresponds to the standard configuration used in this guide.
- The second file, "`lufp1 – exemple du tutorial fedc32p tsx57252.stx` ", is a PL7 PRO file and provides the example for a TSX Premium PLC with a TSX 57252 processor where the LUFP1 is configured under PL7.
- The third file, "`lufp1_tutorial_en_fedc32p_tsx57353.stx`", is a PL7 PRO file and provides the example for a TSX Premium PLC with a TSX 57353 processor where the LUFP1 is configured under PL7.
- The fourth file, "`lufp1_tutorial_en_fedc32_tsx57252.stx`", is a PL7 PRO file and provides the example for a TSX Premium PLC with a TSX 57252 processor where the LUFP1 is configured under AbcConf. The configuration "`LUFP1 - FED C32 - Exemple.cfg`" needs to be loaded onto the LUFP1.
- The fifth file, "`lufp1_tutorial_en_fedc32_tsx57353.stx`", is a PL7 PRO file and provides the example for a TSX Premium PLC with a TSX 57353 processor where the LUFP1 is configured under AbcConf. The configuration "LUFP1_FEDC32_Example.cfg" needs to be loaded onto the LUFP1.

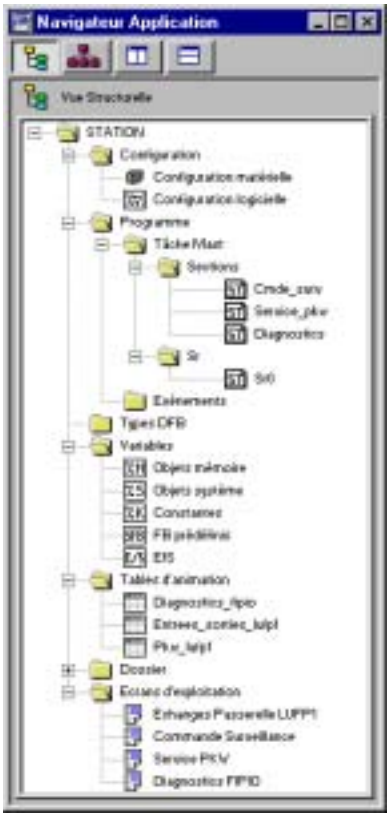
The content and application of these examples is dealt with in the following sections.

The configuration of the first file corresponds exactly to what is described in the previous sections; its content is therefore not detailed here. However, the PL7 PRO file is described below, in terms of the structure of the program sections used and the related operating screens.

11.1. Overview of the "LUFP1_tutorial" example

In this example, the various program sections and sub-programs (icon ST), and operations screens (icon) are organised as follows:

- Command and control screen, 8 TeSys U motor starters:
  - ST Cmd_mon
  - ST Sr0
  - Command Control
- Using the indexed periodic variables (PKW) service to read or write the value of a register:
  - ST Pkw_service
  - PKW Service
- FIPIO and LAS service diagnostic commands:
  - ST Diagnostics
  - FIPIO Diagnostics



Each of the above groupings is described in a separate section.

This description remains concise, because it is only meant to describe the overall operation of the program and the way to use the related screen. If you need a greater degree of detail, don't hesitate to review the actual content of the example under PL7 PRO.

11. Appendix D: Sample Use under PL7 PRO

The source code contains numerous remarks to help you understand how it works. Each "program" file starts with a short description of the way it operates, and each line includes a comment.

Each variable used has a symbol whose name describes its use. Use the keyboard shortcuts “Ctrl+E” and “Ctrl+F” to display the variables by address (e.g. %MW100) or by symbol (e.g. Pkw_checked_boxes_dn).

Three animation tables (icon ) have been created, “FIPIO Diagnostics”, "lufp1_inputs_outputs" and “lufp1_Pkw”. However, the operating screens provided are more than sufficient to present the application data used throughout this example.

11.2. Displaying LUFP1 Gateway Inputs / Outputs

The **“LUFP1 gateway exchanges”** operating screen comprises two separate panes:

- Hexadecimal display of gateway input data (%IW\0.2.2\0.0 to %IW\0.2.2\0.0.31) in the **INPUTS** frame. These inputs are named and grouped in the same way as in this guide (see Section 4.2.7 Configuring Gateway Inputs / Outputs, page 32). Of course, the display of these input data is correct only if the standard gateway configuration is used.

  In the case of the standard configuration, these input data comprise the periodic Modbus data (command and control of TeSys U motor starters), the Modbus aperiodic data (indexed periodic variables / PKW service) and the input word of the list of active slaves (LAS) service, as well as the free and reserved memory locations.

- Hexadecimal display of gateway output data (%QW\0.2.2\0.0 to %QW\0.2.2\0.0.31) in the **OUTPUTS** frame. *As above*.

ENGLISH

11.3. Command and Control of the 8 TeSys U Motor Starters

The **“Command Control”** operating screen (see illustration on next page) allows you to monitor the status of the 8 TeSys U motor starters, numbered from 1 to 8. It also enables you to command them individually by means of a number of buttons.

Registers 455 and 704 on each of the 8 TeSys U motor starters are used to conduct this control and command function:

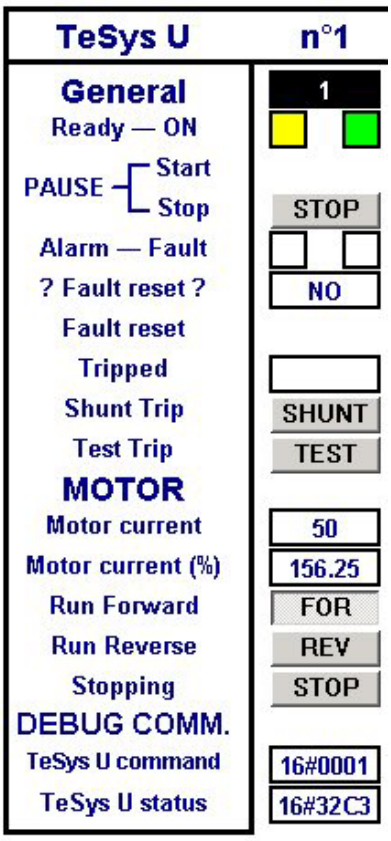
455 – TeSys U status register (IEC61915)
Bit 0 ..... Motor starter ready
Bit 1 ..... Contactor in ON position (1)
Bit 2 ..... Fault (*trip* or *dropout*)
Bit 3 ..... Alarm present
Bit 4 ..... Specific: *Tripped* (I >>)
Bit 5 ..... Specific: Fault reset authorised
Bit 6 ..... Specific: A1-A2 supplied
Bit 7 ..... Specific: Motor running
Bits 8-13 .. Motor current (2#10 0000 = 200%)
Bit 14 ..... Reserved: Local control
Bit 15 ..... Ramping (motor starting)

704 – Command register (IEC61915)
Bit 0 ..... Reserved: Run forward
Bit 1 ..... Reserved: Run reverse
Bit 2 ..... Reserved (stop)
Bit 3 ..... Reset
Bit 4 ..... Reserved (emergency start)
Bit 5 ..... Self test: Triggering test (*trip*)
Bit 6 ..... Reserved (low speed)
Bits 7-11 .. Reserved by IEC61915 standard
Bit 12 ..... Specific: Overload (*shunt trip*)
Bit 13 ..... Specific: Pause (reserved for adjustment)
Bits 14-15 Specific: Reserved

11. Appendix D: Sample Use under PL7 PRO

These statuses and commands are grouped into two sections: “General Status”, for the general operating mode of the motor starters, and "Motor" for that of the controlled motors. The last section, “DEBUG COMM.”, displays the two registers %IW and %QW used for each motor starter.

This screen is shown opposite, but only for the first motor starter, as it is identical to the 7 others.



Most of the displays in this operating screen are directly linked to the %MW registers, which in turn contain the values of registers %IW4.0.1 to %IW4.0.8 and %QW4.0.1 to %QW4.0.8 (status and command registers of the TeSys U motor starters). Only the indirect commands and statuses are described below.

The **“Cmd_mon”** program performs the following tasks:

- It copies the values of words %MW20 to %MW27 into output registers %QW\0.2.2\0.0 to %QW\0.2.2\0.0.7 and copies the values of input registers %IW\0.2.2\0.0 to %IW\0.2.2\0.0.7 into words %MW10 to %MW17.

  These data are copied because word bit extraction can be carried out on %MW-indexed words, but not on %IW-indexed words. The sub-program **“Sr0”** makes extensive use of word indexing, as it can be used to manage any motor starter, the latter being designated by the word %MW0. *E.g.* “%MW10[%MW0]:X13” is authorised, but not “%IW\0.2.2\0.0 [%MW0]:X13”.

  Furthermore, the input and output words are copied one by one, as PL7 PRO does not support expressions such as “%IW\0.2.2\0.0:8”.

- It runs the **“Sr0”** sub-program calling loop to command and control the 8 TeSys U motor starters. At each iteration of the loop, the word %MW0 (“Module”) takes a value from 0 to 7 in order to serve as an index for words %MW10 to %MW17 (inputs) and %MW20 to %MW27 (outputs).

The **“Sr0”** sub-program is called once per motor starter from the **“Cmd_mon”** program. Each of these calls should be made with a different value in the word %MW0 (between 0 and 7) as it is used to index the status word and the command word of the same motor starter. The sub-program is used by the screen shown above to perform the following tasks:

- Determining the overall status of the motor starter with: I >> in case of triggering (*tripped*), 1 if the contactor on the motor starter is in the ON position, and 0 in all other cases.
- Using the commands generated by buttons START and STOP to switch the motor starter into or out of pause mode (bit 13 of the command word). **N.B.** The "pause" mode should not be used for a normal application; use the FOR, REV and STOP buttons instead (see next page).
- Activating the motor starter fault reset command (bit 3 of the command word) if the user pressed the RESET button and the fault LED is on: . Once this command has been activated, it is cancelled as soon as the LED goes off: .
- Activating the motor starter self test command (*trip*) (bit 5 of the command word) if the user pressed the TEST button and the “Tripped” LED is off: . Once the command has been activated, it is cancelled as soon as the LED goes on: .

11. Appendix D: Sample Use under PL7 PRO

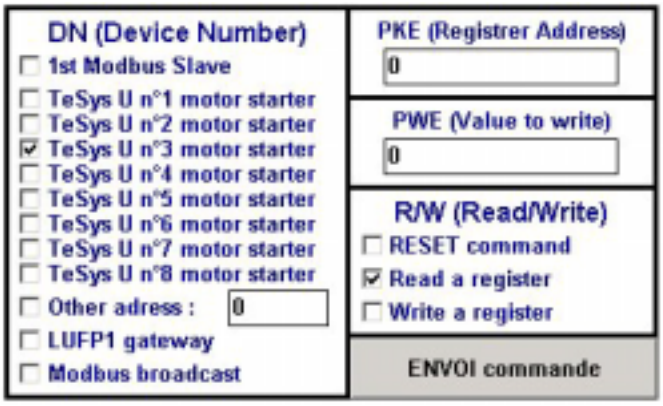
- Evaluating the value of the motor current, given as a percentage of the IR current value (unit = % FLA). Bits 8 (LSB) to 13 (MSB) of the motor starter status word are extracted and the appropriate unit, i.e. 3.125% FLA, is then used to evaluate the current. The maximum value is therefore 63, or 196.875% FLA.

- Using the FOR, REV and STOP buttons exclusively to enable only one of the three following commands at a time, while resetting the other two to zero, in descending order of priority: stop (bit 2 of the command word), run forward (bit 0), and run reverse (bit 1).

11.4. Using the Indexed Periodic Variables (PKW) Service

The **"PKW service"** operation screen allows the user to command the reading or the writing of a register on a Modbus slave, on the gateway, or on all the Modbus slaves at once (broadcast). The screen is divided up into a number of frames, as illustrated below:

- The first frame, "DN (Device Number)", is used to select the station to be polled by the PKW service. Only one box can be checked at any given time. Depending on the box selected when the "SEND command" button is clicked, the DN field (MSB of %QW\0.2.2\0.0.29) of the PKW service command will be updated accordingly (see Section 5.3.2 PKW Service: Command and Response, page 44). The "Other address:" box allows the user to type in an address ranging from 1 to 247.



- The second frame, "PKE (Address Register)", allows you to set the address of the register to be read / written. When the "SEND command" button is clicked, the value currently entered in this frame is copied into the PKE field (%QW\0.2.2\0.0.28) of the PKW service command.

- The third frame, "PWE (Value to be written)", will only be used for write commands on registers. When the "SEND command" button is clicked, the write command is copied into the first word (%Q\0.2.2\0.0.30) in the PWE field of the PKW service command, providing the "Write a register" box is checked.

- The fourth frame, "R/W (Read / Write)", allows you to select the type of command generated by the PKW service: Read, Write or Reset the response. Only one box can be checked at any given time.

- The "SEND command" button triggers the update of the gateway's periodic output data (%QW\0.2.2\0.0.28 to %QW\0.2.2\0.0.31) so that the gateway generates a command that matches the settings and values in the frames above. These updates are performed in the **"Pkw_service"** program. Of course, using periodic data means that a new command can only be generated when one of the values involved changes from one command to the next. This is why the "Reset" command is useful, for example, so that the same command can be repeated several times.

  E.g. In the above example, the frames are configured to issue a read command (16#52) on the register located at address 455 (16#01C7) of TeSys U motor starter n°3 (16#03).

ENGLISH

11. Appendix D: Sample Use under PL7 PRO

- The last frame, beneath and apart from the others, displays the output data transmitted to the gateway in order to generate the corresponding PKW command, and the input data from the gateway's response to this command. The following ***example*** contains the PKW service output and input data for the instances described at the bottom of the previous page. The reading value is 16#01C3.

%QW\p.2.c\0.0.28	16#01C7	PKE - Adresse paramètre	16#01C7	%IW\p.2.c\0.0.28
%QW\p.2.c\0.0.29	16#0352	DN - Device Number (MSB) + R/W - Function Code (LSB)	16#0352	%IW\p.2.c\0.0.29
%QW\p.2.c\0.0.30	16#0000	PWE - Parameter Value (n°1)	16#00C3	%IW\p.2.c\0.0.30
%QW\p.2.c\0.0.31	16#0000	PWE - Parameter Value (n°1)	16#0000	%IW\p.2.c\0.0.31

The **"Pkw_service"** program includes the ST instructions that use the values entered in the **"PKW service"** frame, described on the previous page, to update the PLC outputs for the PKW service command on the gateway (%QW\p.2.c\0.0.28 to %QW\p.2.c\0.0.31). The program handles the following tasks:

- It tests the checkboxes in the "DN (Device Number)" and "R/W (Read / Write)" frames. If, in either frame, none of the boxes is checked, one of them will be selected by default (DN = 1st Modbus slave; R/W = Read a register).
- It compares the values over two consecutive PLC cycles in order to keep only one box checked at any given time in each frame.
- When it detects a click on the "SEND command" button, it updates the local variables, followed by the outputs corresponding to the gateway's PKW service. The purpose of this two-stage update procedure is to isolate the use of the outputs from the rest of the program.

ENGLISH

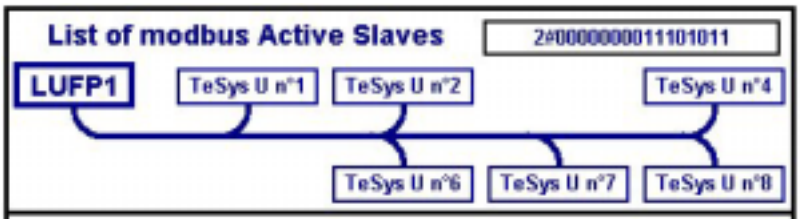
11.5. LUFP1 Gateway Diagnostics

The **"FIPIO Diagnostics"** operating screen provides a graphic example of the use of the list of active slaves (LAS) service, specific to the gateway, and of the FIPIO diagnostic objects that a TSX Premium PLC assigns to the stations on its FIPIO network. See the PL7 PRO user manual or online help for more details about using FIPIO diagnostics on Premium PLCs.

The screen is divided up into three separate frames, as illustrated below:

- **List of active Modbus slaves:** Each "TeSys U n°•" in this frame indicates that the corresponding TeSys U motor starter is present. The hexadecimal value in the top right corner is that of the FIPIO input for the list of active slaves (LAS) service, namely %IW\p.2.c\0.0.27. See Section 5.2 List of Active Slaves (LAS) Service, page 42, for a detailed description of the LAS service.

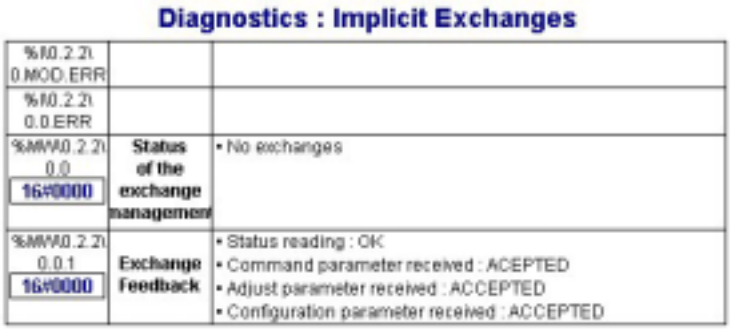
  In this example, bits 2 and 4 of the LAS service input word have been reset by the gateway. Modbus slaves n°3 and 5 are therefore absent.

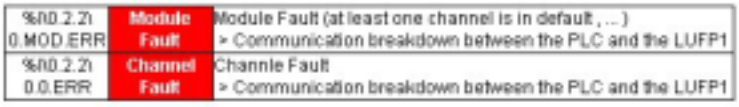


  The "LUFP1" box uses a system bit on the Premium PLC to indicate whether the gateway is present on, or absent from, the FIPIO network (coloured blue or red accordingly). This system bit (%SW128:X2 in this example) is described in Section 5.4.1 System Words %SW128 to %SW135, page 53.

11. Appendix D: Sample Use under PL7 PRO

- **Diagnostics: Implicit exchanges:** The first two rows of the table in this frame correspond to the "Module fault" (%I\p.2.c\0.MOD.ERR) and "Channel fault" (%I\p.2.c\0.0.ERR) described at the beginning of Section 5.4 FIPIO Diagnostic Objects, page 52. The last two rows correspond to the words %MW\p.2.c\0.0 and %MW\p.2.c\0.0.1, both described in Section 5.5 Status of Explicit Exchanges, page 54.





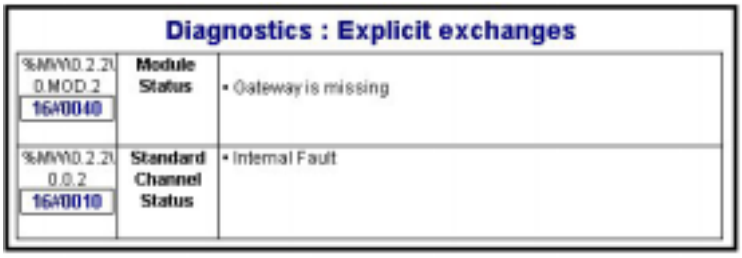
  If the FIPIO connection between the PLC and the gateway is lost, the first two rows signal the presence of just such a fault (see example).

  If these faults arise, the "Explicit exchanges", refreshed every 5 seconds (see below), indicate the presence of an "Internal fault" (Standard channel status) and "Gateway absent" (Module status). A sample screen is shown below.

- **Diagnostics: Explicit exchanges:** Both rows of the table in this frame are refreshed every 5 seconds. They indicate the nature of any errors detected either by the gateway or by the FIPIO master.

  See Sections 5.4.2 Channel Status (%MW\p.2.c\0.0.2), page 53, and 5.4.3 Module Status (%MW\p.2.c\0.MOD.2), page 53, for a description of the explicit objects introduced here.

  The example opposite illustrates one type of error (gateway out of supply).



ENGLISH

The **"Diagnostics"** program includes all the ST instructions for generating the explicit FIPIO diagnostic commands. The program handles the following tasks:

- A timeout (%TM4) is used to generate two explicit diagnostic commands every 5 seconds. One of these commands reads the "Module status" and the other reads the "Standard channel status".

- A second timeout (%TM5) is used to prolong the "Read status in progress" alert (%MW\p.2.c\0.0:X0) by 500 ms, so that it can be seen in the "Exchange management status" line of the **"FIPIO Diagnostics"** screen. This indicates the activity triggered by the two explicit diagnostic commands (see previous point).

12. Appendix E: Modbus Commands

Only the Modbus commands shown in the table opposite are supported by the gateway. The structure of the query and response frames for each of these commands is described in the rest of this chapter.

Function code		Broadcast (1)	Modbus command
3	16#03	—	Read Holding Registers
6	16#06	Yes	Preset Single Register
16	16#10	Yes	Preset Multiple Registers

(1) The content of this column shows whether the command can be added ("Yes") or not ("—") to the list of commands for a broadcaster node, known as "Broadcaster" in AbcConf.

In the following sections, each byte in the query and response frames of a Modbus command is described, in sequence, with the exception of the fields opposite: these are always present in the queries and responses of Modbus commands.

The first two bytes of these frames correspond to the "Slave Address" and "Function" fields. The last two bytes are the two "Checksum" bytes.

Slave Address	- Non-modifiable (Modbus address: 1 to 247. Addresses 125, 126, and 127 prohibited)
Function	- Non-modifiable (Modbus command code)
… Other fields…	*… Specific Modbus command data…*
Checksum (Lo)	- Type of error check
Checksum (Hi)	- Number of the 1st byte checked

The following descriptions of the Modbus frames are mainly intended to help you to configure the gateway's Modbus exchanges using AbcConf. See the documentation of each Modbus slave to check for any restrictions on the use of these frames (number of registers which can be read or written in a single Modbus command, for example).

It is preferable to get hold of a standard Modbus document, such as the *Modicon Modbus Protocol Reference Guide* (ref. PI-MBUS-300 Rev. J), to see how the elements displayed in AbcConf map match the content of the corresponding Modbus frames. Here is a mapping example for a complete frame (including the start and end of frame fields shown above), based on the "Read Holding Registers" Command (16#03) (Section 12.1, page 121):

	Elements under AbcConf		Modbus frame fields	Size
Modbus query	Slave Address		Slave no.	1 byte
	Function		Function no.	1 byte
	Starting Address (Hi, Lo)		No. of 1st word (MSB / LSB)	2 bytes
	Number of points (Hi, Lo)		Number of words (MSB / LSB)	2 bytes
	Checksum		CRC16 (LSB / MSB)	2 bytes
Modbus response	Slave Address		Slave no.	1 byte
	Function		Function no.	1 byte
	Byte count		Number of bytes read	1 byte
	Data		Value of 1st word (MSB / LSB)	2 bytes
			……	……
			Value of last word (MSB / LSB)	2 bytes
	Checksum		CRC16 (LSB / MSB)	2 bytes

Section 7.11 Adding and Setting Up a Modbus Command, page 80, also shows a few examples of mapping between the elements displayed in AbcConf and the corresponding Modbus frame fields.

N.B. Here, the notions of "input" and "output" (and assimilated concepts) are meaningless, in as far all Modbus commands have access to the entire memory of the Modbus slave. We continue to use these terms, however, in order to conform to the terminology of the standard Modbus documentation.

12. Appendix E: Modbus Commands

12.1. "Read Holding Registers" Command (16#03)

Frame	Field	Value or properties
	Starting Address (MSB)	- Address of the 1st output / internal register
	Starting Address (LSB)	
	Number of points (MSB)	- Number of output / internal registers
	Number of points (LSB)	
Response	Byte count	- Number of data bytes = Number of output / internal registers x 2
	Data (first register / MSB)	- Byte swap = "No swapping" (or "Swap 2 bytes") - Data length = Value of the "Byte count" field - Data location = Address in the gateway's input memory
	Data (first register / LSB)	
	……	
	Data (last register / MSB)	
	Data (last register / LSB)	

12.2. "Preset Single Register" Command (16#06)

Frame	Field	Value or properties
Query	Register (MSB)	- Address of the output / internal register
	Register (LSB)	
	Preset data (MSB)	- Byte swap = "No swapping" (or "Swap 2 bytes") - Data length = 16#0002 - Data location = Address in the gateway's output memory
	Preset data (LSB)	
Response	Register (MSB)	- Byte swap = "No swapping" (or "Swap 2 bytes") - Data length = 16#0002 - Data location = Address in the gateway's input memory **N.B.** These data are an echo to the query: in most cases there is no need to feed them back to the FIPIO master.
	Register (LSB)	
	Preset data (MSB)	
	Preset data (LSB)	

Instead of placing the echo of the response to the "Preset Single Register" Command (16#06) in the addresses reserved for FIPIO inputs (16#0000-16#0033), you can put it at 16#0400. This is what the gateway does automatically when you configure periodic command words.

12.3. "Preset Multiple Registers" Command (16#10)

Frame	Field	Value or properties
Query	Starting Address (MSB)	- Address of the 1st output / internal register
	Starting Address (LSB)	
	No. of Registers (MSB)	- Number of output / internal registers
	No. of Registers (LSB)	
	Byte Count	- Number of data bytes = Number of output / internal registers x 2
	Data (first register / MSB)	- Byte swap = "No swapping" (or "Swap 2 bytes") - Data length = Value of the "Byte count" field - Data location = Address in the gateway's output memory
	Data (first register / LSB)	
	……	
	Data (last register / MSB)	
	Data (last register / LSB)	
Response	Starting Address (MSB)	- Address of the 1st output / internal register
	Starting Address (LSB)	
	No. of Registers (MSB)	- Number of output / internal registers
	No. of Registers (LSB)	

ENGLISH

12. Appendix E: Modbus Commands

12.4. Modbus Protocol Exception Responses

When it cannot process a command issued by a Modbus query, a slave sends an exception response instead of the normal response to the query.

With standard Modbus commands, the LUFP1 gateway considers that all the exception responses it receives from Modbus slaves are incorrect responses. As a result, it will carry out the retransmissions configured for the queries concerned.

The structure of an exception response is independent of the Modbus command associated with the “Function” field of the query involved. The complete frame of an exception response is shown below:

Slave Address	Modbus address (1 to 247; addresses 125, 126 and 127 prohibited): The value of this field is identical to that of the “Slave Address” field in the query involved.
Function	Command code, with exception indicator: The value of this field is set to 16#80 + the value of the “Function” field in the query involved.
Exception Code	Code indicating the nature of the error which has caused the exception response (see table below).
Checksum (Lo)	Error check
Checksum (Hi)	

Code	Exception	Description of the exception
16#01	ILLEGAL FUNCTION	The query’s “Function” command is not implemented in the Modbus slave software, or it is unable to process it for the moment.
16#02	ILLEGAL DATA ADDRESS	The combination of the query’s “Starting Address” and “No. of Registers” fields (or assimilated fields) gives access to one or more addresses which are not accessible on the Modbus slave.
16#03	ILLEGAL DATA VALUE	The value of one of the Modbus query’s fields is outside the authorised range. This error does not affect the content of the “Data” (or assimilated) fields, as it only governs the fields used in managing the Modbus protocol.
16#04	SLAVE DEVICE FAILURE	An unrecoverable failure has occurred when processing the command.
16#05 (1)	ACKNOWLEDGE	The Modbus slave informs the gateway that it has accepted the command (acknowledgement), but that it will take too long to process it and it cannot afford to wait for the completion of this process before sending a response. The gateway will need to transmit further queries in order to determine whether the command has finished.
16#06 (1)	SLAVE DEVICE BUSY	The Modbus slave informs the gateway that it is already running a command and cannot therefore run the one transmitted to it. The gateway will need to retransmit the query later.
16#07 (1)	NEGATIVE ACKNOWLEDGE	The Modbus slave informs the gateway that it cannot process the requested command. This exception only affects commands 13 and 14 (16#0D and 16#0E). These functions are not part of the standard Modbus commands and are not described in this document.
16#08 (1)	MEMORY PARITY ERROR	The Modbus slave informs the gateway that it has detected a parity error on access to its own memory. This exception only affects standard commands 20 and 21 (16#14 and 16#15), which are not supported by the gateway.

(1) See the standard Modbus documentation for further information about these various scenarios.

LUFP1 User Manual **V1.1**

2003-05